ComSIS
Computer Science and Information Systems
Volume 19, Number 2, June 2022

Contents

ISSN: 1820-0214 (Print) 2406-1018 (Online)

Computer Science and Information Systems
Vol 19, No 2, June 2022

ComSIS

Computer Science and Information Systems

Published by ComSIS Consortium

Volume 19, Number 2
June 2022

ComSIS is an international journal published by the ComSIS Consortium

ComSIS Editorial Office:
University of Novi Sad, Faculty of Sciences,
Department of Mathematics and Informatics
Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia
Phone: +381 21 458 888; **Fax:** +381 21 6350 458
www.comsis.org; Email: comsis@uns.ac.rs

Volume 19, Number 2, 2022
Novi Sad

Computer Science and Information Systems

ISSN: 1820-0214 (Print) 2406-1018 (Online)

The ComSIS journal is sponsored by:

Ministry of Education, Science and Technological Development of the Republic of Serbia
http://www.mpn.gov.rs/

Computer Science and Information Systems

AIMS AND SCOPE

Computer Science and Information Systems (ComSIS) is an international refereed journal, published in Serbia. The objective of ComSIS is to communicate important research and development results in the areas of computer science, software engineering, and information systems.

We publish original papers of lasting value covering both theoretical foundations of computer science and commercial, industrial, or educational aspects that provide new insights into design and implementation of software and information systems. In addition to wide-scope regular issues, ComSIS also includes special issues covering specific topics in all areas of computer science and information systems.

ComSIS publishes invited and regular papers in English. Papers that pass a strict reviewing procedure are accepted for publishing. ComSIS is published semiannually.

Indexing Information

ComSIS is covered or selected for coverage in the following:

- Science Citation Index (also known as SciSearch) and Journal Citation Reports / Science Edition by Thomson Reuters, with 2021 two-year impact factor 1.170,
- Computer Science Bibliography, University of Trier (DBLP),
- EMBASE (Elsevier),
- Scopus (Elsevier),
- Summon (Serials Solutions),
- EBSCO bibliographic databases,
- IET bibliographic database Inspec,
- FIZ Karlsruhe bibliographic database io-port,
- Index of Information Systems Journals (Deakin University, Australia),
- Directory of Open Access Journals (DOAJ),
- Google Scholar,
- Journal Bibliometric Report of the Center for Evaluation in Education and Science (CEON/CEES) in cooperation with the National Library of Serbia, for the Serbian Ministry of Education and Science,
- Serbian Citation Index (SCIndeks),
- doiSerbia.

Information for Contributors

The Editors will be pleased to receive contributions from all parts of the world. An electronic version (LaTeX), or three hard-copies of the manuscript written in English, intended for publication and prepared as described in "Manuscript Requirements" (which may be downloaded from http://www.comsis.org), along with a cover letter containing the corresponding author's details should be sent to official journal e-mail.

Criteria for Acceptance

Criteria for acceptance will be appropriateness to the field of Journal, as described in the Aims and Scope, taking into account the merit of the content and presentation. The number of pages of submitted articles is limited to 20 (using the appropriate LaTeX template).

Manuscripts will be refereed in the manner customary with scientific journals before being accepted for publication.

Copyright and Use Agreement

Computer Science and Information Systems

Volume 19, Number 2, June 2022

CONTENTS

Special Section: Intelligent systems and their applications

Computer Science and Information Systems 19(2): i–iv https://doi.org/10.2298/CSIS220200iI

Contemporary Research Trends in Computer Science and Informatics – Editorial

Mirjana Ivanović, Miloš Radovanović, and Vladimir Kurbalija

University of Novi Sad, Faculty of Sciences
Novi Sad, Serbia
{mira,radacha,kurba}@dmi.uns.ac.rs

In this second issue of Computer Science and Information Systems for 2022, we are happy to announce the impact factors of our journal, updated for 2021: the new two-year IF is 1.170, and the five-year IF 0.922. We would like to thank all our productive authors and diligent reviewers, whose work in challenging and exciting areas carries the impact of our journal. We hope to continue in the same direction and that the current issue will offer our readers interesting articles in contemporary and emerging research areas.

This issue consists of 22 regular articles and 2 articles in the special section containing selected and extended versions of papers published in proceedings of the International Conference on INnovations in Intelligent SysTems and Applications (INISTA) 2021. We are once again grateful for the hard work and enthusiasm of our authors and reviewers, without which the current issue, as well as the publication of the journal itself, would not have been possible.

The first regular article, "Neural Coreference Resolution for Slovene Language" by Matej Klemen and Slavko Žitnik kicks off this issue by introducing a coreference resolution dataset for Slovene language comparable to English-based corpora. The article also presents a series of analyses using various models from simple linear ones to current state-of-the-art deep neural coreference approaches, investigating robustness of the models using cross-domain data and data augmentations, thereby justifying the introduction of the new corpus with respect to an already existing smaller data set.

The second article, "A Novel Security Mechanism for Software Defined Network Based on Blockchain" by Xian Guo et al. tackles the problems of centralized software defined fetwork (SDN) schemes by proposing a security framework for SDN based on Blockchain (BCSDN) which adopts a physically distributed and logically centralized multi-controller architecture. Simulation of the new scheme is implemented on the Mininet network emulation platform, with experiments performed to verify the solution.

In "Reasoning on the Usage Control Security Policies Over Data Artifact Business Process Models, " Montserrat Estañol et al. propose an enrichment of the standard business process model Notation (BPMN) with a UML class diagram to describe the data model, that is also combined with security policies defined using the UCON ABC framework, with the goal to provide a context where more complex reasoning (for model verification and business process validation) can be performed. This is achieved by integrating the original models, including security policies, into the BAUML artifact-centric business process modeling framework.

Guojiang Shen et al., in their article "Enhancing Interactive Graph Representation Learning for Review-based Item Recommendation" propose IGRec, a new recommendation model enhancing interactive graph representation learning for review-based item recommendation by combining information about users, items and reviews in a single

graph, fuzing edge information into nodes, apply the multilayer graph convolutional network to learn the high-order interactive information of nodes, obtain the final embedding of users/items, and adopt the factorization machine to complete the rating prediction.

"A Study on Optimally Constructed Compactly Supported Orthogonal Wavelet Filters" authored by Yongkai Fan et al., designs compactly supported orthogonal wavelet filters, in which both the scaling and wavelet functions have many vanishing moments, by approximately solving a system of nonlinear equations using proposed optimization algorithms for the Gauss-Newton type method that expand the selection range of initial values.

The article "Eye Movement Analysis in Simple Visual Tasks" by Kiril Alexiev and Teodor Vakarelsky proposes two approaches for noise cancellation in eye-tracker signals and two approaches for microsaccade detection. The obtained results can be a good starting point for interpretation by neurobiologists about the causes of different types of movement and their dependence on the individuality of the observed person and the specific mental and physical condition.

Andrea Stevens Karnyoto et al., in "Transfer Learning and GRU-CRF Augmentation for Covid-19 Fake News Detection" applied BERT and GPT2 as pre-trained using the BiGRU-Att-CapsuleNet model and BiGRU-CRF feature augmentation to solve the fake news detection problem in the Constraint @ AAAI2021 – COVID19 Fake News Detection in English data set. Experimental results show that the hybrid models outperform the non-hybrid baseline, and that BERT consistently outperformed GPT2, achieving accuracy of over 90%.

"Performance and Scalability Evaluation of a Permissioned Blockchain Based on the Hyperledger Fabric, Sawtooth and Iroha" by Arnold Woznica and Michal Kedziora compares different Blockchain platform implementations: Hyperledger Iroha implementing YAC consensus, Sawtooth implementing the PoET algorithm, and the Hyperledger Fabric framework implementation. Various parameters were varied and average transaction latency, network throughput, and transaction failure rate used as measures for evaluation. The results shed light on the impact of a particular parameter on the private blockchain network performance and show how they can be adjusted to improve performance.

In "PE-DCA: Penalty Elimination Based Data Center Allocation Technique Using Guided Local Search for IaaS Cloud," Sasmita Parida et al. propose an approach to locate suitable data centers (DCs) with reduced cost, response time, and processing time for particular user requests by taking into consideration that other requests should not be penalized in terms of time and cost. The approach, Penalty Elimination-based DC Allocation (PE-DCA), addresses, computes, and eliminates the penalties involved in the cost and time through iterative technique using the defined objective and guide functions.

Fatima Zohra Merabet and Djamel Benmerzoug, in "QoS Prediction for Service Selection and Recommendation with a Deep Latent Features Autoencoder," propose a novel framework named auto-encoder for neighbor features (Auto-NF) for predicting quality of service (QoS) values and reduce prediction error. The approach consists of three steps: extended similarity computation method to compute the similarity between users, form clusters of similar neighbors and partition the initial matrix into sub-matrices based on these clusters to reduce the data sparsity problem, and build a simple autoencoder that can learn deep features and select an ideal number of latent factors to reduce the overfitting.

"ProRes: Proactive Re-Selection of Materialized Views" by Mustapha Chaba Mouna et al. first presents a concise state of the art of the materialized view selection problem (VSP) in the database field, and then propose a proactive re-selection approach that considers three important query properties concurrently: largescale queries, query dynamicity, and high query interaction. Extensive experiments are conducted using the Star Schema Benchmark data set to evaluate the effectiveness and efficiency of the approach.

Milan Segedinac et al. in "A Neuroevolutionary Method for Knowledge Space Construction" propose a novel method for the construction of knowledge spaces based on neuroevolution, where knowledge states are considered as neurons in a neural network. The main advantage of the proposed approach is that it is more suitable for constructing large knowledge spaces than other traditional data-driven methods.

The article "Hyper-graph Regularized Subspace Clustering With Skip Connections for Band Selection of Hyperspectral Image" by Meng Zeng et al. proposes a novel clustering method for band selection of hyperspectral image. The approach, hyper-graph regularized subspace clustering with skip connections (HRSC-SC), combines subspace clustering into a convolutional autoencoder by treating it as a self-expressive layer. Symmetrical skip connections are added to the networks to pass image details from encoder to decoder in order to tackle the problem of vanishing gradients.

In "Optimized Placement of Symmetrical Service Function Chain in Network Function Virtualization," Nhat-Minh Dang-Quang and Myungsik Yoo address the problem of efficiently finding suitable placement of virtual network functions (VNFs) in network function virtualization (NFV) when linking the VNFs together as a service function chain (SFC) in situations when SFCs have a complex structure. This is achieved by formulating VNF placement as an optimization problem with symmetrical SFCs that can support both symmetric and asymmetric traffic flows.

Zeyu Sun et al., in "MEC-MS: A Novel Optimized Coverage Algorithm with Mobile Edge Computing of Migration Strategy in WSNs" propose a novel optimized coverage algorithm with mobile edge computing of migration strategy (MEC-MS) with the goal of reducing the overall number of sensor nodes. Experimental results show that the average number of working sensor nodes in the MEC-MS algorithm is 9.74% lower than that of two baseline algorithms, and the average value of network coverage is 9.92% higher.

"Recent Advancements in Privacy-Aware Protocols of Source Location Privacy in Wireless Sensor Networks: A Survey" by Pradeep Kumar Roy et al. is a review article that summarises the protocols proposed in recent research on secure location information in wireless sensor networks (WSNs). Source location privacy (SLP) is an area that attracts a lot of research attention, which a large number of solutions are provided for it. An up-to-date survey of the field does not currently, which is a gap addressed by this article.

The article "RG-SKY: A Fuzzy Group Skyline Relaxation for Combinatorial Decision Making" by Sana Nadouri et al. proposes to extend group skyline dominance by making it more demanding so that several groups leave incomparable. Then, the original group skyline will be enlarged by some interesting groups that are not much dominated by any other group. This is achieved by introducing a new fuzzy preference relation named "much preferred."

In "An Approach to Email Categorization and Response Generation," Sasa Arsovski et al. present the personal email responder (PER): a novel system for email categorization and semi-automatic response generation, whose key novelty is in the approach to email

categorization that distinguishes query and non-query email messages using natural language processing (NLP) and neural network (NN) methods.

Manjie Zhai et al., in their article "A Consortium Blockchain-Based Information Management System For Unmanned Vehicle Logistics" design and implement a system based on the hyperledger fabric blockchain platform to address the risks of order data leakage and tampering in the intelligent logistics distribution environment employing unmanned vehicle delivery. Experimental results show that the proposed system can maintain high throughput in a large-scale request environment under the premise of ensuring data security.

"A Dockerized Big Data Architecture for Sports Analytics" authored by Yavuz Melih Özgüven et al. describes a big data architecture based on Docker containers with Apache Spark, and evaluates the architecture on four data-intensive case studies in sport analytics, including structured analysis, streaming, machine learning approaches, and graph-based analysis.

Charbel Obeid et al., in "A Novel Hybrid Recommender System Approach for Student Academic Advising Named COHRS, Supported by Case-based Reasoning and Ontology" propose hybrid a hybrid recommender system (RS) approach named COHRS that incorporates the knowledge base (KB) and collaborative filtering (CF) recommender techniques, in the domain of student academic advising. Experimental evaluation demonstrates high accuracy of COHRS based on two criteria: the accuracy of retrieving the most similar cases and the accuracy of generating personalized recommendations.

The final regular article "A Machine Learning Approach for Learning Temporal Point Process" authored by Andrija Petrović et al. proposes a novel methodology for learning temporal point processes based on one-dimensional numerical integration techniques. These techniques are used for linearising the negative maximum likelihood (neML) function and enabling backpropagation of the neML derivatives.

Computer Science and Information Systems 19(2): v–vi https://doi.org/10.2298/CSIS220200vK

Guest Editorial – Intelligent systems and their applications

Zeynep Hilal Kilimci, Serdar Solak and Süleyman Eken

Information Systems Engineering, Kocaeli University
Umuttepe Campus, Izmit 41001, Kocaeli, Turkey
{zeynep.kilimci,serdars,suleyman.eken}@kocaeli.edu.tr

International Conference on INnovations in Intelligent SysTems and Applications (INISTA) 2021 aims to bring together the researchers from the entire spectrum of the multi-disciplinary fields of intelligent systems and to establish effective means of communication between them. In particular, it focuses on all aspects of intelligent systems and the related applications, from the points of view of both theory and practice. From around three submitted papers to this particular section, two papers were selected based on the reviews. Each paper was reviewed by at least two reviewers and went through at least two rounds of reviews. The brief contributions of these papers are discussed below.

Keziban Günce Orman, "Dynamic Network Modelling with Similarity based Aggregation Algorithm". Modeling complex systems correctly allows for the finding of hidden knowledge that cannot be explored using standard approaches. The author concentrated on two fundamentally interconnected problems of dynamic network: determining the appropriate/ideal temporal window size for dynamic network snapshots and obtaining a proper dynamic network model using this size. Experiments were realised on four simple or complex data sets by comparing proposed methodology with baseline approaches. According to experiments, compression ratios can extract more noise-free and informative networks than baseline techniques. Furthermore, the aggregation approach has reduced noise levels even further without jeopardizing the overall and critical properties of the system.

Traian Lavric, Emmanuel Bricard, Marius Preda, and Titus Zahari, "A Low-Cost AR Training System for Manual Assembly Operations". The authors proposed a low-cost AR training system for a manual assembly process in a boiler-manufacturing factory. They discussed the design and the implementation of the proposed AR authoring tool, dedicated to shop floor experts for capturing assembly knowledge in a one-step authoring process. Further, they presented how the captured information was conveyed and consumed via AR, for training purposes by novice workers. During their long-term case study, they discovered that relying on low-cost visual assets like text, image, video, and predetermined supplementary information, rather than CAD data and animations, was the optimal compromise for addressing industrial difficulties and needs. They performed two field tests in a real-world use-case to test their hypothesis. According to the results of the first field experiment, spatially registered 2D low-cost visual assets are sufficient and effective for transferring industrial production experience to beginner workers via AR. In the second field experiment, they compared a CAD-enhanced instruction set to the original (low-cost-based) to see if there were any advantages to transmitting assembly information using non-animated, registered CAD models.

The guest editors hope that the research contributions and findings in this special section would benefit the readers in enhancing their knowledge and encouraging them to work on various aspects of areas of computer science and information systems.

Acknowledgments. We want to express our sincere thanks to the editor-in-chief, Prof. Mirjana Ivanovic, and editorial assistants Dr. Vladimir Kurbalija, Dr. Jovana Vidakovic and Dr. Davorka Radakovic for allowing us to organize this particular section. The editorial office staffs are excellent, and thanks for their support. We are also thankful to all the authors who made this special section possible, and to the reviewers for their thoughtful contributions.

Computer Science and Information Systems 19(2):495–521 https://doi.org/10.2298/CSIS201120060K

Neural Coreference Resolution for Slovene Language

Matej Klemen and Slavko Žitnik

University of Ljubljana, Faculty of Computer and Information Science
Večna pot 113, 1000 Ljubljana
{matej.klemen, slavko.zitnik}@fri.uni-lj.si

Abstract. Coreference resolution systems aim to recognize and cluster together mentions of the same underlying entity. While there exist large amounts of research on broadly spoken languages such as English and Chinese, research on coreference in other languages is comparably scarce. In this work we first present SentiCoref 1.0 - a coreference resolution dataset for Slovene language that is comparable to English-based corpora. Further, we conduct a series of analyses using various complex models that range from simple linear models to current state-of-the-art deep neural coreference approaches leveraging pre-trained contextual embeddings. Apart from SentiCoref, we evaluate models also on a smaller coref149 Slovene dataset to justify the creation of a new corpus. We investigate robustness of the models using cross-domain data and data augmentations. Models using contextual embeddings achieve the best results - up to 0.92 average F_1 score for the SentiCoref dataset. Cross-domain experiments indicate that SentiCoref allows the models to learn more general patterns, which enables them to outperform models, learned on coref149 only.

Keywords: coreference resolution, Slovene language, neural networks, word embeddings.

1. Introduction

Coreference resolution is a task where the goal is to identify and group all entity mentions that refer to a common entity in the text. It is an important part of the attempt to understand language at a higher level and has its role across many other tasks in natural language processing. One such example is question answering, where the user can provide a complex query, often mentioning the same entity with different words to construct a less monotone sentence. For the system to determine what the user is asking and respond correctly, it must be able to figure out what the user is referring to across a long span of text.

Generally, the task can be thought of as a combination of mention detection and mention clustering, and many approaches explicitly perform these two steps when doing coreference resolution. The mention detection step deals with the detection of all entities that refer to some entity in the text. Mention clustering then divides the entities into groups based on the entity they refer to.

The most researched languages on this topic include broadly spoken languages such as English and Chinese. However, less-researched (and less-resourced) languages often possess interesting phenomena that do not appear in English and could provide a source

of difficulties for English systems. In our work, we focus on Slovene, an example of such a language, so far being the topic of few analyses.

We experiment with two datasets: coref149 and SentiCoref, with our work being the first analysis performed on the latter. As such, we provide a detailed description of the dataset and compare it with coref149 and some commonly used English coreference resolution datasets. We simplify our analysis, only studying the performance of systems on the mention clustering task and assuming that the system can do the mention detection step sufficiently accurately in advance. We analyze the performance of variously complex models on datasets providing a substantially different amount of resources to learn from. The studied models range in complexity from a simple linear baseline with features described by existing literature, to complex models that use contextual embeddings, pretrained on general multilingual or Slovenian data. Additionally, we study how transferable the patterns learned on both datasets are by either augmenting the datasets or learning a model on one dataset and evaluating its performance on the other. Throughout the analysis, we probe the effect of certain architectural decisions, such as embedding size or the amount of provided context, to additionally examine the capabilities of models and datasets. We complement the quantitative evaluation (using automated metrics) with additional qualitative analysis, outlining common mistakes in the best performing model. The source code for our experiments is available online [1].

The rest of the paper is structured as follows. In Section 2 we provide an overview of existing approaches to coreference resolution. In Section 3 we describe the datasets used in our experiments, with additional focus on SentiCoref. In Section 4 we describe the methods we use in our experiments. We then present and analyze the empirical results in Section 5. Finally, in Section 6, we summarize our work and provide some possible directions for further research.

2. Related Work

Coreference resolution is a widely studied problem in computational linguistics. Anaphoras and coreferent entities form a subset of discourse parsing [1] which is crucial for text understanding. A discourse is a group of interrelated sentences that contribute to a clear understanding only when read together. Anaphora on the other hand represents references (i.e. mentions) to items mentioned earlier in discourse. The primary anaphora type is the pronominal anaphora [2]. In contrast to anaphora, coreference identifies words or phrases (i.e. mentions) referring to an underlying unique entity. Most coreference resolution systems deal with two tasks: (a) mention detection and (b) mention clustering. As mention detection could be heuristically solved to identify mention candidates based on part-of-speech tagging, most systems focus on solving mention clustering. The latter is also the main focus of our work.

2.1. Coreference Resolution in English

Most approaches in coreference resolution transform the problem into a binary classification problem, where the goal is to determine whether two selected mentions are coref-

[1] https://github.com/matejklemen/slovene-coreference-resolution

erent or not [3,4]. Prior to the use of deep learning approaches, methods based on conditional random fields [5] and rule-based methods [6] were achieving state-of-the-art results. The problem with such approaches is that they treat all coreference candidates independently, so they cannot choose the most probable candidate when multiple valid ones exist. Mention ranking was introduced as an improvement over those methods [7]. In these approaches, candidates for coreference are scored using a score and the best scoring candidates are selected as coreferent ones. The benefit of such an approach is that it does not consider candidates in isolation but jointly with other mentions. Another improvement is the entity-mention approach [7], where the models are trained to determine whether the observed mention belongs to one of the coreference clusters [8].

Recently, Lee et al. [9] introduced and evaluated the effectiveness of an end-to-end approach for coreference resolution, where the steps of mention detection and clustering are trained jointly using deep neural networks. They introduce a span ranking approach and optimize the two steps jointly by factoring the coreference compatibility score between two spans i and j into a part that models how likely it is that the two spans are actual mentions and a part that models how likely span j is an antecedent of span i. A potential problem of such an approach is that there are a lot of candidate spans to consider, which is solved by pruning the space of candidate spans. The method considers only a portion of the top N spans, selected based on the score that models how likely span i is a mention. In later work, the same authors [10] introduce another part to the coreference compatibility score that roughly models how likely span j is an antecedent of span i and use it to prune the candidate space even further. Subsequent work includes modifications such as the use of more sophisticated contextual embeddings [11] or more specialized ones [12] inside the end-to-end system. The latter work introduces SpanBERT, a modified version of Bidirectional Encoder Representations from Transformers (BERT) [13], which introduces a span masking and a span boundary objective as customized optimization objectives, designed to help span modeling tasks, such as coreference resolution.

2.2. Coreference Resolution in Non-English Languages

Due to the ubiquity of the English language and the availability of resources, the majority of work on coreference resolution is focused on the analysis of English data. However, studies exist for a wide variety of languages, presenting approaches that use rules, classic machine learning techniques or deep neural networks. Early approaches for various languages often tend to rely heavily on rules. Examples of such approaches include various systems in Polish [14], Lithuanian [15] and Russian [16]. These approaches offer a good starting point due to being well-studied and showing promising results in different languages. They are also relatively transparent, which enables their use in specific domains. For example, the Lithuanian approach performs coreference resolution on medical data.

After using rule-based systems, there was a shift towards using machine learning models combined with hand-engineered features. A positive aspect of such approaches is that there exist common features that work well across different languages, although they might have different importance. However, the features can automatically be weighted and combined by the models. This claim is supported by literature which adapts English systems and applies them to another language, such as the Polish adaptation [17] of Beautiful Anaphora Resolution Toolkit (BART) [18] as well as in the existing literature for Slovene language [19], where Žitnik and Bajec analyze the effectiveness of a wide

range of features, previously proposed for English. Similarly, a baseline approach in our work uses proven features in combination with a linear model and is shown to perform well across both Slovene datasets but still worse than the approaches using deep learning.

Lately, the approaches for languages other than English are also starting to shift towards the use of deep learning. Park et al. [20] use word embeddings and a feed-forward neural network to model coreference resolution as a binary classification problem and show its effectiveness for the Korean language, while Nitoń et al. [21] experiment with deep learning approaches that use a combination of word embeddings and handcrafted features and either a fully-connected neural network or a Siamese network [22] in a mention ranking or entity-mention approach. Training deep neural networks typically requires a large dataset to tune the weights stably. For some languages, annotated resources are either not available or very scarce, which is one of the reasons why authors experiment with learning cross-lingual coreference resolution. For example, Urbizu et al. [23] present a coreference resolution system for the Basque language, which they train on an English corpus. They compare the cross-lingual system with a monolingual (Basque) one and show that the cross-lingual system works slightly better. Similarly, although motivated by language similarity instead of data scarcity, Cruz et al. [24] present a coreference resolution system for Portuguese, which they learn on a Spanish corpus. They are able to achieve competitive performance to a monolingual system, trained on Portuguese.

Our work draws inspiration from existing literature and studies it in terms of the Slovene language. To the best of our knowledge, there currently exist no Slovene coreference resolution systems based on deep learning. In addition to this, our work is the first to analyze coreference resolution systems on the SentiCoref dataset [25].

3. Coreference Resolution Datasets

The majority of the state-of-the-art systems were evaluated on specialized shared tasks at MUC (Message Understanding Conference) [26], ACE (Automatic Content Extraction) [27], SemEval2010 (Semantic Evaluation) [28], and at CoNLL-2011 and CoNLL-2012 (Conference on Computational Language Learning) [29,30]. Nowadays, datasets presented at these shared tasks or conferences still represent the main coreference resolution benchmark datasets. Recently, some specific coreference resolution datasets were produced, such as gender-focused coreference resolution [31], commonsense-related coreference resolution [32] and coreference resolution as a part of general language understanding dataset [33].

In our experiments we use two Slovene coreference resolution datasets: coref149 [19], containing 149 documents, and SentiCoref [25], containing 837 documents. First, we provide some general statistics for both datasets and compare them to commonly used English datasets. Then, as our work presents the first analysis on SentiCoref, we provide a more detailed description of the dataset in Section 3.1.

We provide general statistics for both used datasets in Table 1. In addition, we note statistics for some other commonly used English datasets. We can see that coref149 is comparably small to the other datasets, being composed of less documents and containing less tokens. On the other hand, SentiCoref 1.0 dataset contains more documents than ACE 2004 and SemEval2010 which seems promising for training coreference resolution

models for Slovene. Most of the corpora (except coref149) are made up of news documents.

Table 1. Dataset statistics for the Slovene (coref149 and SentiCoref 1.0) and most often used English (ACE 2004, SemEval2010 and CoNLL-2012) coreference resolution datasets.

Statistic	coref149	SentiCoref 1.0	ACE 2004	SemEval2010	CoNLL-2012
Documents	149	837	450	314	2,135
Tokens	26,960	433,139	191,387	102,952	1,468,889
Entities	1,277	14,572	12,439	20,921	37,330
Trivial	831	7,721	-	-	-
Mentions	2,329	42,738	29,724	28,242	174,437
Overlapping	196	4,212	-	-	-

Interestingly, the ratio of tokens per document is similar among all datasets. The number of entities per document is comparable between SentiCoref 1.0 and CoNLL-2012, while it is lower for ACE 2004 and SemEval2010. Such rough comparison can provide an initial insight into whether SentiCoref 1.0 dataset is on par with the commonly used English datasets.

It is important to notice that there are a number of differences between the Slovene and English language. Apart from the fact that Slovene is a highly inflected language, it introduces verb as a new mention type. In Slovene texts, references to entities are often implicitly hidden in verbs and not mentioned explicitly as in English. Due to annotation specifics (which we describe in more detail in Section 3.1), we also report the number of trivial entities and overlapping mentions. Trivial entities contain only one mention in a document, while overlapping mentions are mentions that overlap in tokens, although they can refer to different entities. For example, the text "*Slovenian football club Olimpija*" contains three mentions ("*Slovenian*", "*Olimpija*" and "*Slovenian football club Olimpija*"), which refer to two entities (Slovenia and football club Olimpija).

3.1. SentiCoref 1.0 Dataset

In this section, we provide a more detailed description of SentiCoref 1.0, a dataset that was created to enable Slovene coreference resolution experiments on a larger scale. It is publicly available online [25].

For SentiCoref 1.0 we selected 837 articles from the existing SentiNews 1.0 corpus [34] which consists of 10,427 manually annotated Slovenian news articles for sentiment analysis. The content represents online news related to politics, business, economics and finance. The news were randomly sampled from Slovenian online news portals 24ur, Dnevnik, Finance, RTVSLO and Žurnal24. In SentiNews, each article is independently annotated by between two and six annotators for sentiment analysis using a five-level Lickert scale (very negative, negative, neutral, positive and very positive) on three levels of granularity (document, paragraph and sentence level). For SentiCoref, we selected documents from SentiNews that contain between 50 and 73 named entities, as detected

by Polyglot [35]. In Figure 1 we show a part of an annotated document from the dataset. It contains three types of annotations, which we describe next.

Named entity annotation: The basis for coreference resolution and target-level sentiment analysis are entities. In the corpus we therefore focused only on entities that contain at least one named entity mention in a document. This means that entities never explicitly mentioned in the corpus are not taken into account (e.g. if the entity is always referred to using pronouns). Based on the existing Slovene named entity recognition dataset [36] we decided to annotate:

- **(a) persons or groups of persons:** For example [Alfred Nobel], [poslanec SKD] (eng. parliament member from the SKD party) or [zamejci] (eng. Slovenes abroad).
- **(b) organizations:** For example [Švedska centralna banka] (eng. Swedish Central Bank). This category also includes political parties, for example [SKD] (SKD party).
- **(c) geographical names:** For example locations, such as [Maribor] and [Washington], political geographical units, such as [EU].

Coreference resolution annotation: Coreferences are annotated only for entities that contain at least one named entity mention in a document and represent identity-level coreferences. Thus, each coreference chain refers only to one specific underlying entity and not, e.g., a part-whole concept.

Target-level sentiment analysis annotation: One of the aims of the dataset was also to provide sentiment annotation for each entity in a document. As an entity is represented as a list of coreferent mentions, the task is to identify the sentiment of an entity in the context of a document. So, if there is a description of a crime that a person committed, then such entity would be annotated as a negative entity. Annotations for the entities are added to the last mention of an entity in a document.

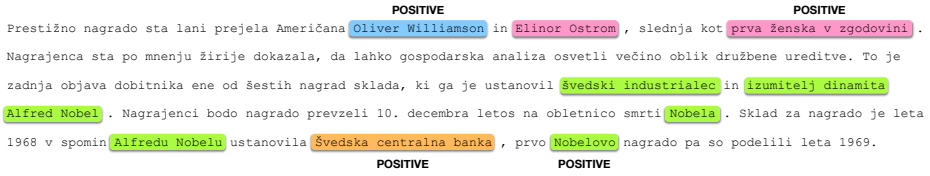


Fig. 1. Part of an annotated document from SentiCoref. Each entity and its coreferences are marked with the same color. Sentiment annotation is marked at the last mention of an entity. The English translation of the text is: "*The prestigious award went to Americans Oliver Williamson and Elinor Ostrom, the latter being the first woman in history to receive it. According to the jury, the winners have shown that economic analysis can shed light on most forms of social regulation. This is the latest announcement of the winner of one of six awards given by the organization, founded by the Swedish industrialist and inventor of dynamite Nobel. The winners will receive the prize on 10. December on the anniversary of Nobel's death. The Prize Fund was established in 1968 in memory of Alfred Nobel by the Swedish Central Bank, with the first Nobel Prize being awarded in 1969.*"

In Table 2 we show the general statistics of named entity types and sentiment values. Note that there is a difference of 451 entities between Table 2 and Table 1. This is the number of entities that do not have a sentiment value annotation. The lack of annotations was discovered after the end of the annotation campaign.

Table 2. Number of entities by their type and sentiment in the SentiCoref 1.0 dataset.

	Positive	**Neutral**	**Negative**	**All**
Person	637	2,611	542	3,790
Organization	756	4,455	986	6,197
Location	274	3,603	257	4,134
All	1,667	10,669	1,785	14,121

The dataset was annotated by a total of eight different annotators, with each document being annotated by two different annotators. All the documents were then manually curated by the second author of this paper. Compared to English datasets, SentiCoref 1.0 contains the following specifics.

- It contains annotations for overlapping mentions, the number of which we provide in Table 1. These can appear as mentions of different entities or the same. The latter are mostly left predicate complements (premodifiers), for example, "*[head of engineering [Zoran Arnež]*$_1$*]*$_1$" contains two overlapping mentions referring to the same entity. On the other hand, in case of right predicate complements (i.e. postmodifiers), there is always a character between the two mentions, such as '-', 'v', '(', '/' or ',. Such apposition is for example "*[Zoran Arnež]*$_1$*, [head of engineering]*$_1$."
- In Slovene, the mentions can implicitly be hidden inside a verb. In such cases, we annotate part of verbs that contain information about the entity. Such an example is the text "*[Postal je]*$_1$ *učitelj*", which would be directly translated into English as "*[Became]*$_1$ *a teacher*", although it is implied that the statement is about a man. These annotations exist only in cases where no explicit mention of an entity exists in a sentence. Another example is shown in Figure 2.

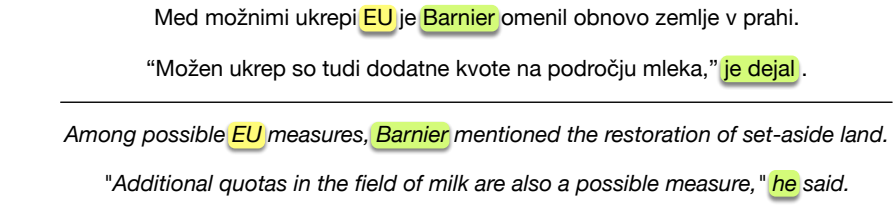


Fig. 2. An example of a Slovene coreference where a coreferent mention is "hidden" within a verb. The figure shows two entities and three mentions. The bottom part is the English translation of the Slovene example.

In Figure 3 we show the part of speech tag distribution in SentiCoref 1.0. We used Stanza [37] to annotate the corpus automatically. In the case of a multi-word mention, we take the type of the first word as the tag of the mention. We observe that nouns are most common, also because named entities are nouns. The next are adjectives which often play the role of a premodifier of a mention. The third and fourth are verbs and pronouns. Compared to English, verbs in Slovene implicitly contain pronoun mentions which are always explicit in English, so these would be represented as one group in English datasets. Other part of speech types are rare and represent special cases that appear at the beginning of mentions, for example, titles ("dr. Lahovnik") or abbreviations ("B. Bonnaud").

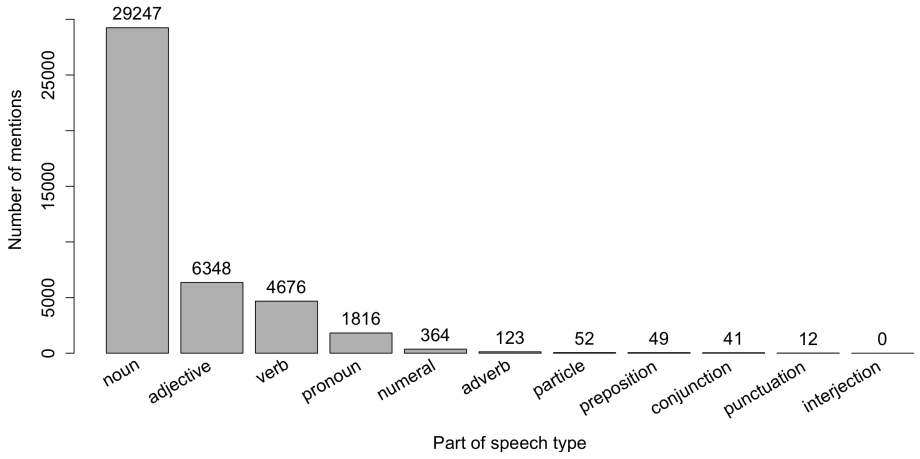


Fig. 3. Distributions of part of speech types of first words of the mentions in SentiCoref.

Lastly, we show three additional distributions for SentiCoref 1.0 in Figure 4: entity size, document size and distance between coreferent mentions. As described in Table 1, there are a lot of trivial entities in the dataset (around 50%). Still, entities containing up to 10 mentions are well represented and mostly contain other half of entities. The distribution of distances between two consecutive coreferent mentions (upper right) is important as it explains the maximum possible performance of a coreference resolution model that can take up to N consecutive mentions as input. For example, distance 0 means that mentions are directly consecutive (no other mentions in between), and distance 1 means that there is one other mention in between. We can observe that by collecting mentions up to a distance of 10 we could address most of the existing coreferences (around 95%).

As we selected only documents that contain at least 5 named entities, the minimum number of mentions per document is larger than that (i.e. 13 mentions). From Figure 4 we observe that most of the documents contain between 30 and 70 mentions. There exist documents with up to 145 mentions, but these are less frequent. These documents are typically sports game reports where a number of players and sports clubs are mentioned.

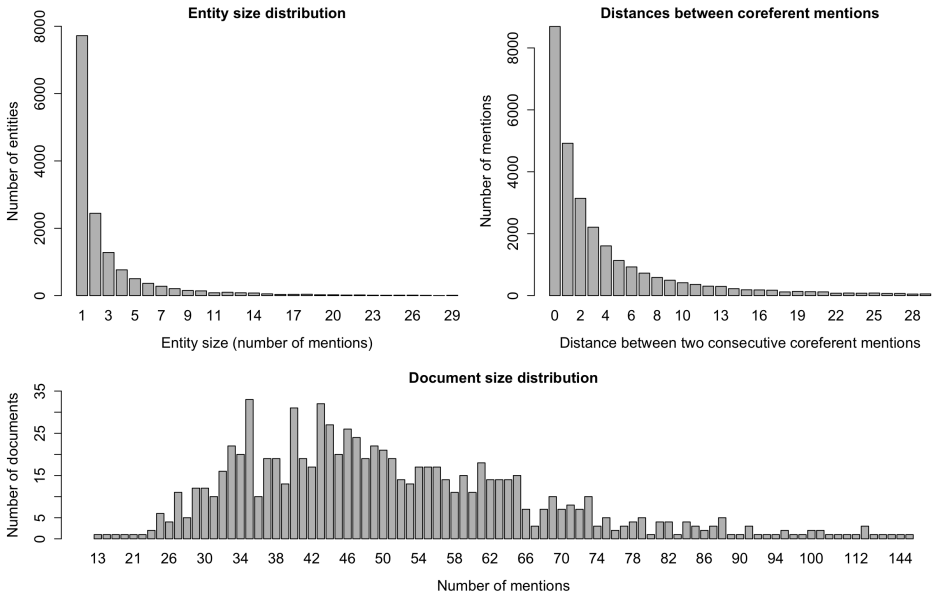


Fig. 4. Distributions of entity sizes in SentiCoref based on the number of mentions (upper left), distances between consecutive coreferent mentions (upper right) and document sizes based on the number of mentions (bottom).

4. Methods

4.1. Mention Ranking Formulation of the Task

In all of our approaches, we treat coreference resolution as a mention ranking problem. We are given a document with information about which spans of words (mentions) refer to the same entity. We move through the mentions in the order of their appearance in the document. For every mention, we determine which preceding mention (antecedent) it is coreferent with. This is done by assigning a coreference compatibility score to all candidates and selecting the mention with the highest score among them as the coreferent mention. Figure 5 shows an example of a mention ranking algorithm.

The goal of the models is to make the coreference compatibility score high for coreferent mentions and low for non-coreferent mentions. Formally, the models minimize the cross-entropy between predicted and the ground truth antecedent probability distribution.

4.2. Baseline Model

Our baseline model is a linear mention pair scorer based on handcrafted features. Scores are obtained for every antecedent candidate appearing in the document and then normalized using the softmax function. For constructing the features, we use additional metadata such as part of speech tags and lemmas. For coref149, this metadata is provided in the

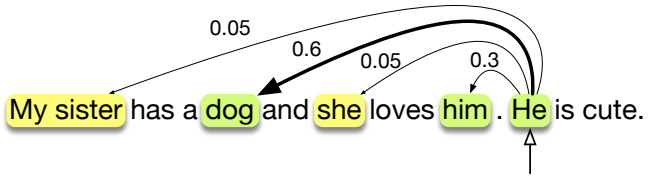


Fig. 5. Mention ranking algorithm. Marked words represent mentions of two different entities, split by color, based on the entity they reference. Mention currently being processed is "He". We compute scores for all antecedent mentions. The mention with the highest score is selected as a coreference.

ssj500k dataset, while for SentiCoref, this metadata is not provided, so we obtain it automatically using the Stanza library [37]. The features we use in our baseline model are based on already-proven ones reported in existing literature [5]. They are described in Table 3. Categorical features are encoded into binary ones using one-hot encoding. In the following sections, we refer to this approach as *linear baseline*.

Table 3. Features used in our linear baseline model.

Feature	Description
string match	exact match for pronouns or match in lemmas
same sentence	are both mentions in same sentence
same gender	one-hot encoded vector for values: same gender, different gender
same number	one-hot encoded vector for values: match in number, don't match in number
is appositive	both mentions have noun-related tag and previous mention is followed by comma
is alias	one mention is a subset of another
is prefix	one mention is prefix of another
is suffix	one mention is suffix of another
is reflexive	one mention is followed by another that is reflexive pronoun
jw dist	distance value between two mentions according to Jaro-Winkler metric

4.3. Neural Models

In this section, we first describe the used neural coreference scoring architecture. We describe it by detailing the process of obtaining the coreference score for a given mention and a coreference candidate. Next, we present our three variations of the architecture, which differ in the type of embeddings, used to represent the mention tokens.

Our neural architecture follows the neural network-based scorer, originally introduced as part of an end-to-end system for coreference resolution [9]. The scorer is shown schematically in Figure 6 and described next.

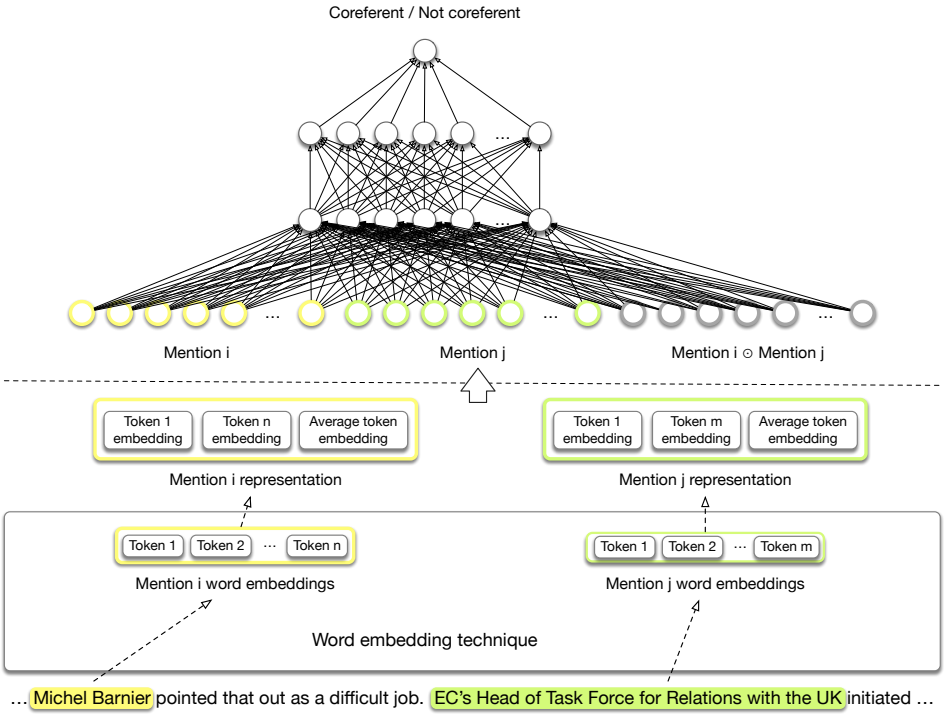


Fig. 6. Neural coreference scorer architecture. Input to the scorer represents both mention representations and their element-wise product. Mention representations consist of first mention token embedding, last mention token embedding and average token embedding of mention tokens.

The input to the scorer are tokens for a mention, and one of the candidate mentions for coreference, while the output is a coreference compatibility score between two mentions, representing how likely it is that the two mentions are coreferent. First, the tokens are embedded using one of the embedding types described later in this section. Then, a three-part mention representation is constructed independently for each mention. This is done by concatenating the embedding of the first token of a mention, the embedding of the last token of a mention and a learned weighted combination of embeddings for all mention tokens. The first and second parts of the representation are used to capture the left and right context of a mention, while the third part is used as an approximate representation of the head word inside a mention. Once the mention representations are obtained for both mentions, a three-part mention pair representation is constructed by concatenating the representations of the first mention, the second mention and their element-wise product. Finally, this is fed into a two hidden layer feedforward neural network with rectified linear

unit (ReLU) activation function to produce a coreference compatibility score, which is then used in the mention ranking framework, described in Section 4.1.

One aspect of the neural architecture, which is still vaguely described, are the embeddings used to represent the tokens. We experiment with different types of embeddings to produce three variations of the previously defined architecture. Specifically, we use non-contextual (word2vec and fastText), contextual ELMo (Embeddings from Language Models) and contextual BERT embeddings. The process of obtaining these embeddings is shown schematically in Figure 7 and described next.

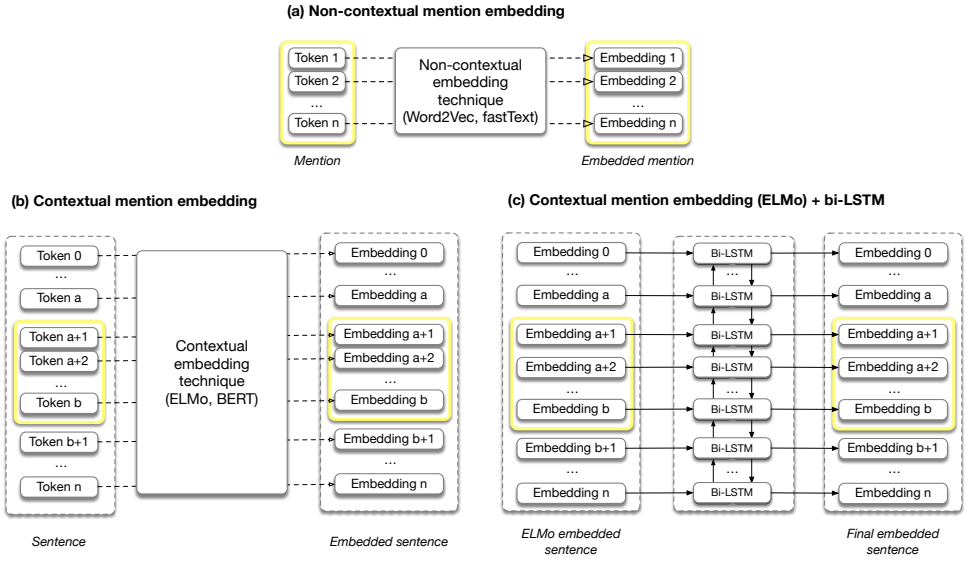


Fig. 7. Different embedding techniques used in our work. The input mention is marked in yellow. The figure shows methods to get (a) non-contextual word embeddings such as word2vec and fastText, (b) contextual word embeddings such as ELMo and BERT, and (c) the additional step that is used for processing ELMo embeddings: a pass through an additional bi-LSTM. For BERT-based embeddings (b), we use the output of its last hidden layer as embeddings.

Non-contextual Embeddings For experiments with non-contextual embeddings, we use word2vec embeddings [38] and fastText embeddings [39], which we provide in their original form as the input to the coreference scorer. Specifically, we use word2vec embeddings trained with the skip-gram architecture and fastText embeddings trained using the continuous bag of words architecture, a decision which we make based on the fact that such embeddings are already provided online. In our primary experiments, we use 100-dimensional word2vec embeddings [40] and 100-dimensional fastText embeddings, which we additionally fine-tune for coreference resolution. As we are dealing with datasets of very different sizes that might not both allow the learning of complex models,

we also experiment with a smaller (50) and bigger (300) dimensionality of word embeddings. For tokens appearing in the vocabulary that do not have an associated pretrained word embedding, we randomly initialize their embeddings to random $[0, 1)$ vectors of used dimensionality.

In the following sections we refer to these approaches as *word2vec* and *fastText*.

Contextual Embeddings: ELMo In the first approach using contextual embeddings, we use Embeddings from Language Models (ELMo) [41]. Following the setup used by the authors of ELMo, we learn a task-specific linear combination of the three ELMo layers. Additionally, we encode the resulting embedded document tokens using a bidirectional LSTM [42], processing each sentence independently. We use a pretrained Slovene ELMo model [43], whose weights we fine-tune together with the weights of coreference scoring module. In the following sections, we refer to this approach as *elmo-lstm*.

Contextual Embeddings: BERT In the second approach using contextual embeddings, we use BERT embeddings [13], following the setup described in existing literature [11], where BERT-embedded tokens are given as input to the coreference scorer. Because BERT has an effective maximum input length, we divide the longer documents into non-overlapping segments of pre-determined maximum length and embed them independently. The embeddings we input to the coreference scorer correspond to the last hidden layer of BERT. To perform batched coreference score computation, we pad the mentions to a fixed maximum span size. Mentions which are longer than the maximum size are truncated. The size is set in a way that most mentions do not get truncated. We use two types of BERT: a trilingual BERT model (CroSloEngual BERT) [44] and multilingual BERT. In the following sections, we refer to these approaches as *CroSloEngual BERT* and *multilingual BERT*.

5. Results and Discussion

In this section, we first explain the experimental settings and metrics used in our coreference resolution experiments. Then, we present the analysis of results obtained by our approaches on Slovene datasets.

5.1. Experimental Framework

There is no general agreement on which metric to use for the coreference resolution task. We adopt the most commonly used measures in the literature, which are described below. Prior to the measures we use in this paper, a graph-based scoring algorithm had been used that produced very unintuitive results [45,46]. There have been several metrics proposed, so we evaluate the system using the following most commonly used measures:

MUC The key idea in developing the MUC measure [47] was to give an intuitive explanation of the results for coreference resolution systems. It is a link-based metric (it focuses on pairs of mentions) and is the most widely used. MUC counts false positives by computing the minimum number of links that need to be added to connect

all the mentions referring to an entity. On the other hand, recall measures how many of the links must be removed so that no two mentions referring to different entities are connected in the graph. Thus, the MUC metric gives better scores to systems with more mentions per entity while ignoring entities with only one mention (singleton entities).

BCubed The BCubed metric [48] (B3) tries to address the shortcomings of MUC by focusing on mentions, and measures the overlap of the predicted and true clusters by computing the values of recall and precision for each mention. If k is the key entity and r the response entity containing the mention m, the recall for mention m is calculated as $\frac{|k \cap r|}{|k|}$, and the precision for the same mention, as $\frac{|k \cap r|}{|r|}$. This score has the advantage of measuring the impact of singleton entities, and gives more weight to the splitting or merging of larger entities.

CEAF The goal of the CEAF metric [49] is to achieve better interpretability. The result reflects the percentage of correctly recognized entities. We use entity-based metric (in contrast to a mention-based version) that tries to match the response entity with at most one key entity. For CEAF, the value of recall is $\frac{\text{total similarity}}{|k|}$, while precision is $\frac{\text{total similarity}}{|r|}$.

We report on precision, recall and F_1 score For each metric. Results are computed using *neleval*[2] package.

In addition, we also report on the **CoNLL 2012** score, which is the average F_1 score of the three metrics (i.e., MUC, B3 and CEAF) and is intended to serve as a compact summary of the model's performance. It was also used during CoNLL 2012 shared task [29] to rank participating coreference resolution systems. Unless noted otherwise, we use this metric to determine if method M_1 is better than method M_2.

We compute the described metrics using different evaluation techniques. On coref149, we use 10-fold cross-validation (CV), meaning we divide the dataset into 10 parts, train a model on 9 folds and evaluate it on the remaining fold. We repeat this 10 times, each time evaluating on a different fold, and report the mean score (along with the standard deviation) across the folds as the final result of a method. On SentiCoref, we instead decide to use a single split into a training, validation and test set in ratio 70%:15%:15%. We choose to do so primarily due to the substantially larger size of the dataset, which reduces the random fluctuation in the performance of the models. The validation set is used to select the best hyperparameters for our model as well as for regularization. The best model is selected with early stopping: once the loss on the validation set does not decrease for 5 consecutive epochs, the training is stopped, and the best state is used for evaluation. In each iteration of CV, an internal 3-fold CV is used in place of a validation set for hyperparameter and model selection.

5.2. Empirical Comparisons

The results achieved by presented methods are shown in Table 4 for coref149 and Table 5 for SentiCoref. Besides our baseline scorer and variations of a neural coreference scorer, we also include results obtained by two trivial models, which show what kind of scores

[2] Neleval package repository: `https://github.com/wikilinks/neleval` (Accessed on: April 9, 2021)

Table 4. MUC, B3 and CEAFe F1 scores of our approaches on the **coref149 dataset**, ordered by average F1 score. The numbers represent the means and standard deviations across 10 folds of CV.

Model	MUC	B3	CEAFe	Avg. F1
All-in-one	0.617 (0.070)	0.358 (0.046)	0.152 (0.029)	0.376 (0.047)
Each-in-own	0.000 (0.000)	0.688 (0.049)	0.562 (0.062)	0.417 (0.037)
fastText100	0.125 (0.090)	0.707 (0.041)	0.589 (0.050)	0.473 (0.043)
word2vec100	0.342 (0.099)	0.670 (0.100)	0.565 (0.113)	0.525 (0.048)
elmo-lstm	0.4246 (0.080)	0.7131 (0.038)	0.645 (0.042)	0.594 (0.035)
linear-baseline	0.539 (0.092)	0.793 (0.043)	0.701 (0.060)	0.678 (0.058)
multilingual BERT	0.719 (0.049)	0.841 (0.038)	0.801 (0.047)	0.787 (0.043)
CroSloEngual BERT	0.720 (0.081)	0.839 (0.033)	0.806 (0.031)	0.788 (0.039)

Table 5. Achieved MUC, B3 and CEAFe F1 scores of our approaches on the **SentiCoref dataset**.

Model	MUC	B3	CEAFe	Avg. F1
Each-in-own	0.000	0.525	0.389	0.305
All-in-one	0.770	0.231	0.050	0.350
linear-baseline	0.605	0.691	0.565	0.620
word2vec100	0.708	0.705	0.658	0.690
fastText100	0.778	0.773	0.753	0.768
elmo-lstm	0.855	0.819	0.810	0.828
multilingual BERT	0.923	0.891	0.886	0.900
CroSloEngual BERT	0.939	0.916	0.912	0.922

one can expect by default: the "Each-in-own" model puts each mention in its own cluster, while the "All-in-one" model puts all mentions of a document into a single cluster. Comparing these methods in isolation, we can see that the former has a higher average score on coref149, while the latter has a higher score on SentiCoref, which agrees with the statistics of trivial entities presented in Section 3: because coref149 contains a larger proportion of trivial entities, the "Each-in-own" model achieves a slightly higher score there.

Linear baseline achieves an average F1 score of 0.678 on coref149 and 0.620 on SentiCoref. It serves as a relatively strong baseline, beating both methods using non-contextual embeddings and one using contextual embeddings on coref149, where data is scarce. On SentiCoref, its performance is inferior to the mentioned methods since their weights can be more reliably tuned there. The results indicate that simpler methods based on manual feature engineering might be viable when we have a small amount of training data. Another desirable trait of the linear model is our ability to inspect what the model has learned by plotting the feature weights. The learned weights on Figure 8 indicate that string equivalence features (such as string match and suffix indicator) are universally useful, while the importance of some other features differs substantially.

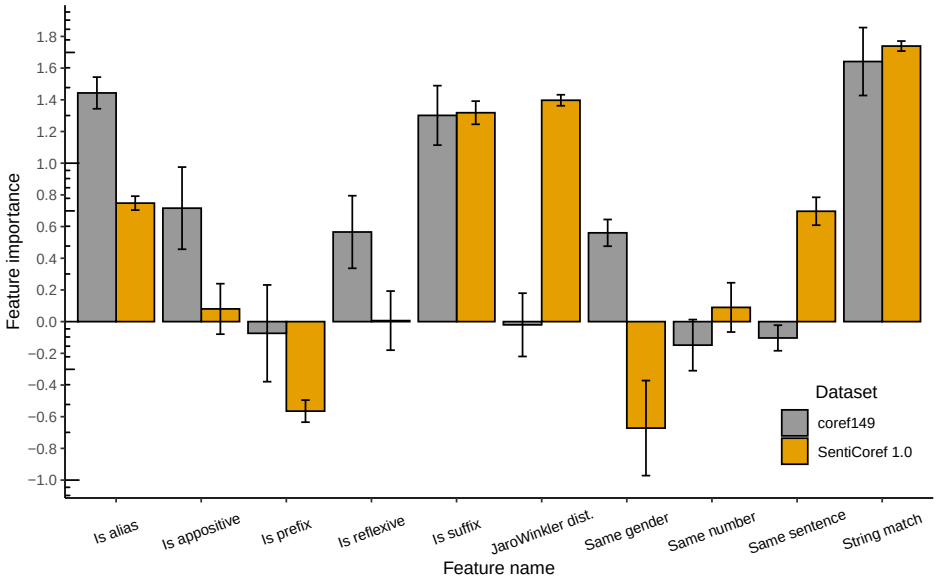


Fig. 8. The weights of linear baseline coreference scorer. For both datasets, the model assigns high importance to the string equivalence based attributes.

Although desirable on small datasets, the results achieved by linear baseline are surpassed by the neural approaches using either non-contextual or contextual embeddings once more data is available, as is the case with SentiCoref.

Focusing on the models using non-contextual embeddings first, we see that they achieve average F1 scores of 0.690 (word2vec) and 0.768 (fastText) on SentiCoref, while they achieve poor average F1 scores of 0.525 (word2vec) and 0.473 (fastText) on coref149. FastText embeddings offer increased flexibility in representing small variations of words due to being based on subword units, though they also amount to a larger amount of trainable weights than word2vec embeddings. The results seem to indicate that the fastText embeddings can not be tuned reliably on the coref149 dataset, so they perform worse than word2vec. Conversely, SentiCoref offers enough data to tune them, which results in a noticeable performance boost (+0.078 average F1 score over word2vec).

The outlined statement is further supported by our experiment with different non-contextual embedding sizes, the results of which are shown in Table 6 for coref149 and Table 7 for SentiCoref. On coref149, the top F1 scores are achieved by smaller embeddings (50-dimensional fastText and 100-dimensional word2vec) since they can be fit most reliably. On SentiCoref, approaches using fastText embeddings of all sizes outperform the approaches using word2vec embeddings. We note however that the two types of embeddings are obtained using different architectures, i.e. skip-gram and continuous bag of words. Some of the outlined differences could also be caused by this, though we do not explore the comparison further.

Table 6. MUC, B3 and CEAFe F1 scores of neural approaches using non-contextual embeddings of different dimensions on the **coref149 dataset**. The numbers represent the means and standard deviations across 10 folds of CV.

Model	MUC	B3	CEAFe	Avg. F1
fastText100	0.125 (0.090)	0.707 (0.041)	0.589 (0.050)	0.473 (0.043)
fastText300	0.132 (0.074)	0.706 (0.040)	0.591 (0.049)	0.477 (0.042)
word2vec50	0.361 (0.111)	0.607 (0.092)	0.479 (0.105)	0.483 (0.046)
word2vec300	0.210 (0.139)	0.680 (0.095)	0.568 (0.102)	0.486 (0.045)
fastText50	0.169 (0.090)	0.711 (0.041)	0.602 (0.059)	0.494 (0.059)
word2vec100	0.342 (0.099)	0.670 (0.100)	0.565 (0.113)	0.525 (0.048)

The results for models using contextual embeddings lead to similar conclusions. On coref149, the models using ELMo embeddings cannot learn many patterns and therefore achieve a poor average F1 score (0.594). This means that even the best score does not surpass the performance of a linear baseline on coref149. Surprisingly, the story is different for BERT models: the multilingual and trilingual BERT model approaches surpass the baseline and achieve a practically equivalent average F1 score (0.787 and 0.788).

Table 7. MUC, B3 and CEAFe F1 scores of neural approaches using non-contextual embeddings of different dimensions on the **SentiCoref dataset**.

Model	MUC	B3	CEAFe	Avg. F1
word2vec50	0.697	0.698	0.642	0.679
word2vec100	0.708	0.705	0.658	0.690
word2vec300	0.733	0.726	0.704	0.721
fastText50	0.768	0.765	0.748	0.761
fastText100	0.778	0.773	0.753	0.768
fastText300	0.785	0.788	0.759	0.777

On SentiCoref, the contextual models can be tuned much better, pushing the achieved scores far above those of non-contextual models and the linear baseline. The model using ELMo embeddings achieves an F1 score of 0.828, while the multilingual and trilingual BERT achieve F1 scores of 0.900 and 0.922, respectively.

The results show the overall effect of using different approaches to model coreference resolution, with the results for BERT on SentiCoref looking particularly impressive. To get additional perspective into the limits of our methods, we qualitatively observe the wrong predictions made by the best performing approach on SentiCoref and point out some error patterns which we observe multiple times, their likely causes and possible solutions. The examples which we refer to in descriptions of error patterns are also shown in Figures 9, 10 and 11.

– **Errors due to limitations of architectural decisions.** One type of error is due to the limited context made available to the BERT model. For example, the model assigns a mention at the end of a long document to a new entity, although the same entity was already detected at the start of the document. In our case, this is likely a consequence of representing the documents as independent segments of a fixed maximum size. The dilemma of how to represent long documents is still an open problem, although one possibility to reduce the number of such errors could be to represent documents as a combination of partially overlapping segments of maximum size, as outlined in work by Joshi et al. [11].
  The second type of error we mention here are the locally consistent but globally inconsistent assignments. For example, consider a document with the following three mentions (Figure 9): “Šrot” (in this case implying a man’s surname), “nadzornik v odvisnih družbah” (meaning “supervisor”) and “Tone Turnšek” (another man’s name and surname). The model first assigns “Šrot” as the antecedent of “nadzornik v odvisnih družbah”. Next, it assigns “nadzornik v odvisnih družbah” as the antecedent of “Tone Turnšek” . Although both of the assignments are potentially valid on their own, they form an inconsistency once both are taken together since the names clearly refer to different persons. The reason for these errors lies in the mention ranking framework, which does not explicitly consider existing entity assignments. Besides

changing the problem formulation, a possible improvement that tries to fix such inconsistencies is the use of an iterative refinement mechanism [10].

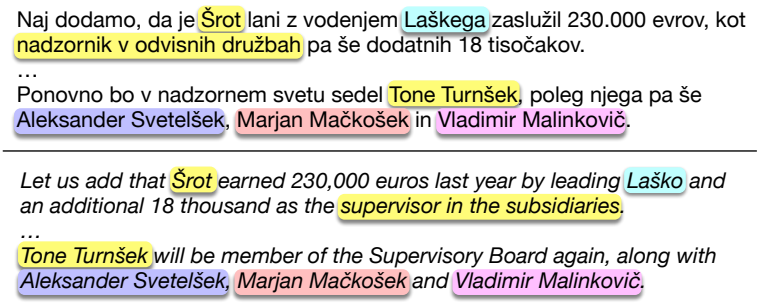


Fig. 9. Example of an error that the best BERT model makes on SentiCoref, likely due to limitations of architectural decisions. "Tone Turnšek" should have been assigned to a separate entity.

– **Lack of common sense.** For example (Figure 10), the model assigns the mentions "Merkur" (a Slovene company) and "nakelski trgovec" (meaning "a retail company based in Naklo") to a different cluster, although the two both refer to the same company. Such situations are arguably challenging even for humans if one does not have the background knowledge, and the modeling of common sense is still an open problem.

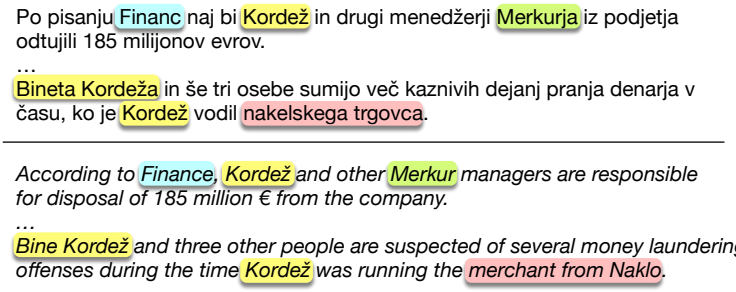


Fig. 10. Example of an error due to lack of common sense by the best BERT model on SentiCoref. "merchant from Naklo" should have been assigned the same cluster as "Merkurja."

– **Assignment of similar, but semantically different, named entities to same cluster.** For example (Figure 11), the mentions "Britanija" (meaning "Britain") and "Brioni" (a group of islands in Croatia) get clustered together, although they refer to two different geographical locations. This may be a consequence of the model putting too much emphasis on the common prefix "Bri" instead of taking into account the entire

words. A possible way to solve this could be to use a gazetteer to divide the mentions into two entities.

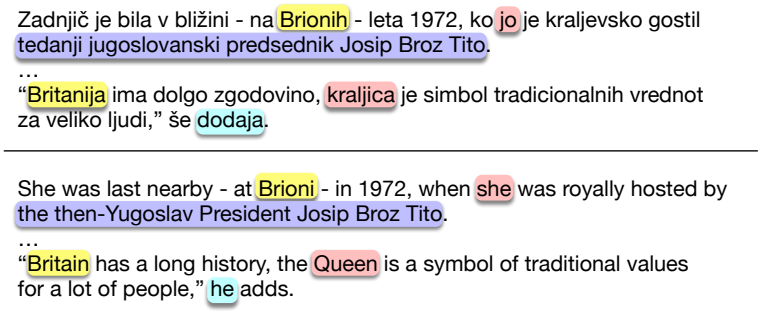


Fig. 11. Example of an error due to assignment of similar, but semantically different mentions by the best BERT model makes on SentiCoref. "Britanija" should have been a separate entity.

The quantitative results show that the highest scores obtained on the two datasets differ significantly. To see whether this gap can be narrowed, we perform additional experiments using augmented datasets. We expand the training subset of one dataset with all examples of the other dataset and rerun the training and evaluation procedure. Additionally, we perform cross-domain experiments, in which we take a model trained on one dataset and evaluate it on the other dataset without additional fine-tuning. The aim of this is to see how transferable the learned patterns are between datasets. We show the results in Table 8 for coref149 and Table 9 for SentiCoref and summarize them next.

The outcome can roughly be divided into two cases. The linear baseline performs equally or worse both with the augmented dataset as well as in cross-domain experiments. As seen in Figure 8, the weights for many features differ substantially between the datasets, so they cannot be set in a way that would benefit both datasets at once. The other approaches generally see a performance increase when using a dataset augmented with SentiCoref, and a comparable or worse performance when using a dataset augmented with coref149. The only model that benefits slightly from the augmentation with coref149 is the ELMo based model. Experimental results show that models trained on SentiCoref or an augmented dataset perform better on coref149 than those trained only on coref149, with the best trilingual BERT model achieving the new highest average F1 score (0.869). This strongly indicates that SentiCoref allows the models to learn more general patterns behind coreference. Therefore its use should be prioritized over coref149.

Throughout our experiments, the results show that once enough data is available, the methods using contextual embeddings (ELMo, BERT) start performing well and learn general patterns behind coreference. In our last set of experiments, we check the effect of certain architectural decisions on the performance of these methods on SentiCoref. Specifically, we observe the effect of three types of modifications:

Table 8. MUC, B3 and CEAFe F1 of our approaches in experiments involving augmented datasets (*augm.*) and cross-domain evaluation (marked as *SentiCoref* as the models are trained only on SentiCoref). The methods are evaluated on **coref149**, and the numbers represent the means and standard deviations across 10 folds of CV.

Model	MUC	B3	CEAFe	Avg. F1
linear-baseline (SentiCoref)	0.303 (0.077)	0.742 (0.044)	0.636 (0.058)	0.560 (0.047)
word2vec100 (SentiCoref)	0.468 (0.069)	0.675 (0.034)	0.589 (0.033)	0.578 (0.033)
word2vec100 (augm.)	0.506 (0.064)	0.713 (0.037)	0.625 (0.048)	0.615 (0.036)
linear-baseline (augm.)	0.491 (0.095)	0.781 (0.044)	0.683 (0.061)	0.652 (0.060)
fastText100 (SentiCoref)	0.539 (0.065)	0.790 (0.030)	0.728 (0.036)	0.686 (0.031)
fastText100 (augm.)	0.572 (0.101)	0.802 (0.042)	0.737 (0.054)	0.704 (0.060)
elmo-lstm (SentiCoref)	0.683 (0.063)	0.819 (0.037)	0.767 (0.042)	0.757 (0.040)
elmo-lstm (augm.)	0.705 (0.097)	0.850 (0.035)	0.816 (0.040)	0.790 (0.048)
multilingual BERT (SentiCoref)	0.787 (0.058)	0.856 (0.039)	0.826 (0.052)	0.823 (0.044)
multilingual BERT (augm.)	0.794 (0.050)	0.882 (0.039)	0.854 (0.050)	0.843 (0.031)
CroSloEngual BERT (augm.)	0.816 (0.073)	0.900 (0.028)	0.876 (0.039)	0.864 (0.043)
CroSloEngual BERT (SentiCoref)	0.826 (0.052)	0.904 (0.030)	0.877 (0.036)	0.869 (0.026)

- Does providing more context to the method using ELMo embeddings bring its performance closer to methods using BERT embeddings? To check this, we replace the independent encoding of sentences with the encoding procedure used in BERT-based models and instead encode non-overlapping segments of 256 words.
- How much does freezing the underlying embeddings and only fine-tuning the remaining layers decrease the performance?
- Does using a learned linear combination of all 12 hidden layers in BERT-based models improve the performance over using only the last hidden state?

The results of modified models are shown in Table 10. First, we can see that providing more context to the ELMo-based model has a negative effect, with its average F1 score decreasing by 0.016 in comparison to the model using a single sentence context. Besides decreasing the performance, the modification also increases the training time as the in-

Table 9. MUC, B3 and CEAFe F1 of our approaches in experiments involving augmented datasets (*augm.*) and cross-domain evaluation (marked as *coref149* as the models are trained only on coref149). The methods are evaluated on **SentiCoref**. Note that models marked with *coref149* are trained using a single 70%:15%:15% data split instead of CV in order to keep the scores comparable.

Model	MUC	B3	CEAFe	Avg. F1
fastText100 (coref149)	0.106	0.538	0.412	0.352
word2vec100 (coref149)	0.350	0.592	0.445	0.462
elmo-lstm (coref149)	0.547	0.546	0.512	0.535
linear-baseline (coref149)	0.611	0.694	0.564	0.623
linear-baseline (augm.)	0.608	0.693	0.568	0.623
word2vec100 (augm.)	0.668	0.703	0.647	0.673
multilingual BERT (coref149)	0.761	0.704	0.666	0.710
CroSloEngual BERT (coref149)	0.764	0.746	0.718	0.743
fastText100 (augm.)	0.783	0.776	0.755	0.771
elmo-lstm (augm.)	0.864	0.830	0.827	0.840
multilingual BERT (augm.)	0.911	0.890	0.885	0.895
CroSloEngual BERT (augm.)	0.921	0.890	0.881	0.897

creased number of words inside a segment means more words are processed sequentially using a LSTM. Second, freezing the underlying embeddings has a noticeable effect on BERT-based models and a small effect on ELMo-based models. The latter is a consequence of the model having an additional LSTM context encoder, which manages to act as a rough replacement for the trainable weights of ELMo. All three variations with frozen embeddings however still outperform the models using non-contextual embeddings. Last, we find that using a learned linear combination of all 12 BERT hidden layers instead of the last hidden layer does not improve the performance further. Multilingual BERT achieves a practically equivalent F1 score of 0.900, while the modified trilingual BERT sees a slight performance decrease with an average F1 score of 0.910.

6. Conclusion

We have introduced a new coreference resolution dataset for the Slovene language and performed experiments on it using variously complex models, showing that it allows us to learn strong models, to the point that they show strong performance even on a different dataset (coref149). Simultaneously, we have evaluated the methods on the existing

Table 10. MUC, B3 and CEAFe F1 scores of our approaches using contextual embeddings with three types of modifications: using 256-word context instead of a single sentence, using a learned linear combination of 12 hidden BERT layers and freezing of underlying embeddings (❄). The methods are evaluated on SentiCoref. For reference, we repeat the results of unmodified approaches (at the beginning).

Model	MUC	B3	CEAFe	Avg. F1
(elmo-lstm)	0.855	0.819	0.810	0.828
(multilingual BERT)	0.923	0.891	0.886	0.900
(CroSloEngual BERT)	0.939	0.916	0.912	0.922
elmo-lstm (segments of 256 words)	0.853	0.802	0.780	0.812
multilingual BERT ❄	0.828	0.799	0.801	0.810
elmo-lstm ❄	0.852	0.806	0.799	0.819
CroSloEngual BERT ❄	0.847	0.813	0.815	0.825
multilingual BERT (12 layers)	0.915	0.896	0.891	0.901
CroSloEngual BERT (12 layers)	0.934	0.903	0.895	0.910

(smaller) dataset and shown that its small size can present a problem for learning more complex models.

Although the best of the analyzed methods show surprisingly good results, it should be noted that we have only tackled the mention clustering part of the problem. The mention detection step undoubtedly introduces some noise to the process. As an example, the authors of the first end-to-end neural coreference resolution system [9] note that the average F1 score of their system increased by 0.175 when they replaced mention detection with oracle mentions. One of the logical next steps would be to check how well an end-to-end approach would work on Slovene data.

Additionally, knowing that the SentiCoref dataset is suitable for learning complex models, a possible next step would be to check if we could use it to aid the learning of coreference resolution for a different language that is similar to Slovene, for example Croatian.

Acknowledgments. The work presented in this paper started as a project in a natural language processing course and was then improved upon with additional experiments. We would like to thank Blažka Blatnik and Martin Čebular for their contributions to this project throughout the course. We would also like to thank the anonymous reviewers for the detailed reviews and helpful comments.

The SentiCoref 1.0 corpus preparation was funded by CLARIN.SI 2019 projects - *Corpus for Slovene coreference resolution and aspect-based sentiment analysis–SentiCoref 1.0.*

References

1. Radu Soricut and Daniel Marcu. Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 228–235, 2003.
2. Shalom Lappin and Herbert J Leass. An algorithm for pronominal anaphora resolution. *Computational linguistics*, 20(4):535–561, 1994.
3. Vincent Ng and Claire Cardie. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 104–111, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.
4. Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544, 2001.
5. Slavko Žitnik, Lovro Šubelj, and Marko Bajec. SkipCor: Skip-mention coreference resolution using linear-chain conditional random fields. *PloS one*, 9(6):e100101, 2014.
6. Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nathanael Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher D Manning. A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 492–501, 2010.
7. Sam Wiseman, Alexander M. Rush, Stuart Shieber, and Jason Weston. Learning anaphoricity and antecedent ranking features for coreference resolution. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1416–1426, Beijing, China, July 2015. Association for Computational Linguistics.
8. Xiaofeng Yang, Jian Su, GuoDong Zhou, and Chew Lim Tan. An NP-cluster based approach to coreference resolution. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 226–232, Geneva, Switzerland, aug 23–aug 27 2004. COLING.
9. Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.
10. Kenton Lee, Luheng He, and Luke Zettlemoyer. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.
11. Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. BERT for coreference resolution: Baselines and analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5803–5808, Hong Kong, China, November 2019. Association for Computational Linguistics.
12. Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020.
13. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

14. Maciej Ogrodniczuk and Mateusz Kopeć. Rule-based coreference resolution module for Polish. In *Proceedings of the 8th Discourse Anaphora and Anaphor Resolution Colloquium (DAARC 2011)*, volume 191, page 200, 2011.
15. Voldemaras Žitkus, Rita Butkienė, Rimantas Butleris, Rytis Maskeliunas, Robertas Damasevicius, and Marcin Woźniak. Minimalistic approach to coreference resolution in Lithuanian medical records. *Computational and Mathematical Methods in Medicine*, 2019:1–14, 03 2019.
16. Svetlana Toldova, Ilya Azerkovich, Alina Ladygina, Anna Roitberg, and Maria Vasilyeva. Error analysis for anaphora resolution in Russian: new challenging issues for anaphora resolution task in a morphologically rich language. In *Proceedings of the Workshop on Coreference Resolution Beyond OntoNotes (CORBON 2016)*, pages 74–83, San Diego, California, June 2016. Association for Computational Linguistics.
17. Mateusz Kopeć and Maciej Ogrodniczuk. Creating a coreference resolution system for Polish. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 192–195, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA).
18. Yannick Versley, Simone Paolo Ponzetto, Massimo Poesio, Vladimir Eidelman, Alan Jern, Jason Smith, Xiaofeng Yang, and Alessandro Moschitti. BART: A modular toolkit for coreference resolution. In *Proceedings of the ACL-08: HLT Demo Session*, pages 9–12, Columbus, Ohio, June 2008. Association for Computational Linguistics.
19. Slavko Žitnik and Marko Bajec. Coreference resolution for Slovene on annotated data from coref149. *Slovenščina 2.0: empirical, applied and interdisciplinary research*, 6(1):37–67, Jun. 2018.
20. Cheoneum Park, Kyoung-Ho Choi, Changki Lee, and Soojong Lim. Korean coreference resolution with guided mention pair model using deep learning. *ETRI Journal*, 38(6):1207–1217, 2016.
21. Bartłomiej Nitoń, Paweł Morawiecki, and Maciej Ogrodniczuk. Deep neural networks for coreference resolution for Polish. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA).
22. Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a "Siamese" time delay neural network. In *Proceedings of the 6th International Conference on Neural Information Processing Systems*, NIPS'93, page 737–744, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.
23. Gorka Urbizu, Ander Soraluze, and Olatz Arregi. Deep cross-lingual coreference resolution for less-resourced languages: The case of Basque. In *Proceedings of the Second Workshop on Computational Models of Reference, Anaphora and Coreference*, pages 35–41, Minneapolis, USA, June 2019. Association for Computational Linguistics.
24. A. F. Cruz, G. Rocha, and H. L. Cardoso. Exploring Spanish corpora for Portuguese coreference resolution. In *2018 Fifth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pages 290–295, 2018.
25. Slavko Žitnik. Slovene corpus for aspect-based sentiment analysis - SentiCoref 1.0, 2019. Slovenian language resource repository CLARIN.SI.
26. Lynette Hirschman and Nancy A. Chincor. MUC-7 coreference task definition. In *Proceedings of the Seventh Message Understanding Conference*, pages 1–17, San Francisco, 1997. Morgan Kaufmann.
27. George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. The automatic content extraction (ACE) program–tasks, data, and evaluation. In *Proceedings of LREC*, volume 4, pages 837—-840, 2004.
28. Marta Recasens, Lluís Màrquez, Emili Sapena, Antònia M. Martí, Mariona Taulé, Véronique Hoste, Massimo Poesio, and Yannick Versley. SemEval-2010 task 1: Coreference resolution in multiple languages. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 1–8, 2010.

29. Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In *Proceedings of the Joint Conference on EMNLP and CoNLL: Shared Task*, pages 1–40. Association for Computational Linguistics, 2012.
30. Sameer Pradhan, Lance Ramshaw, Mitchell Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. CoNLL-2011 shared task: Modeling unrestricted coreference in OntoNotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1—-27, 2011.
31. Sandeep Attree. Gendered ambiguous pronouns shared task: Boosting model confidence by evidence pooling. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 134–146, Florence, Italy, August 2019. Association for Computational Linguistics.
32. Hector J. Levesque, Ernest Davis, and Leora Morgenstern. The Winograd schema challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, page 552–561. AAAI Press, 2012.
33. Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32, pages 3266–3280. Curran Associates, Inc., 2019.
34. Jože Bučar. Manually sentiment annotated Slovenian news corpus SentiNews 1.0, 2017. Slovenian language resource repository CLARIN.SI.
35. Rami Al-Rfou. Polyglot, April 2020.
36. Simon Krek, Kaja Dobrovoljc, Tomaž Erjavec, Sara Može, Nina Ledinek, Nanika Holz, Katja Zupan, Polona Gantar, Taja Kuzman, Jaka Čibej, Špela Arhar Holdt, Teja Kavčič, Iza Škrjanec, Dafne Marko, Lucija Jezeršek, and Anja Zajc. Training corpus ssj500k 2.2, 2019. Slovenian language resource repository CLARIN.SI.
37. Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2020.
38. Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.
39. Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.
40. Andrei Kutuzov, Murhaf Fares, Stephan Oepen, and Erik Velldal. Word vectors, reuse, and replicability: Towards a community repository of large-text resources. In *Proceedings of the 58th Conference on Simulation and Modelling*, pages 271–276. Linköping University Electronic Press, 2017.
41. Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.
42. Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.
43. Matej Ulčar and Marko Robnik-Šikonja. High quality ELMo embeddings for seven less-resourced languages. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4731–4738, May 2020.
44. Matej Ulčar and Marko Robnik-Šikonja. FinEst BERT and CroSloEngual BERT. In *Text, Speech, and Dialogue*, pages 104–111, 2020.

45. Nancy Chincor. MUC-3 evaluation metrics. In *Proceedings of the 3rd conference on Message understanding*, pages 17–24, Pennsylvania, 1991. Association for Computational Linguistics.
46. Nancy Chinchor and Beth Sundheim. MUC-5 evaluation metrics. In *Proceedings of the 5th conference on Message understanding*, page 69–78, Pennsylvania, 1993. Association for Computational Linguistics.
47. Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*, pages 45–52. Association for Computational Linguistics, 1995.
48. Amit Bagga and Breck Baldwin. Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–566. Citeseer, 1998.
49. Xiaoqiang Luo, Abe Ittycheriah, Hongyan Jing, Nanda Kambhatla, and Salim Roukos. A mention-synchronous coreference resolution algorithm based on the Bell tree. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04, page 135–es, USA, 2004. Association for Computational Linguistics.

Matej Klemen is a junior researcher at the University of Ljubljana, Faculty for computer and information science, where he is also a PhD student. He is mainly interested in the development of deep learning approaches for natural language processing. Additionally, he is interested in making the state of the art approaches applicable to languages that are not in the research spotlight, such as Slovene.

Slavko Žitnik is an assistant professor at the University of Ljubljana, Faculty for computer and information science. He is teaching courses related to databases, natural language processing, information retrieval and information systems. He is actively engaged in multiple research projects related to semantic technologies, such as coreference resolution or knowledge base constructions. He is cooperating with a number of research institutions - University of Belgrade, Sorbonné Universite Paris 1, University of South Florida and Harvard University. Apart from research he tries to productivize research results together with the needs of companies and this paper is a research result of such collaborations.

Received: November 20, 2020; Accepted: October 30, 2021.

Computer Science and Information Systems 19(2):523–545 https://doi.org/10.2298/CSIS210222001G

A novel Security Mechanism for Software Defined Network Based on Blockchain

Xian Guo, Chen Wang, Laicheng Cao, Yongbo Jiang, and Yan Yan

School of Computer and Communication,
Lanzhou University of Technology, Gansu 730050, China
iamxg@163.com, {572957016, 28140795, 670342320, 414696390}@qq.com,

Abstract. The decoupling of the data plane and the control plane in the Software-Defined Network (SDN) can increase the flexibility of network management and operation. And it can reduce the network limitations caused by the hardware. However, the centralized scheme in SDN also can introduce some other security issues such as the single point of failure, the data consistency in multiple-controller environment and the spoofing attack initiated by a malicious device in the data plane. To solve these problems, a security framework for SDN based on Blockchain (BCSDN) is proposed in this paper. BCSDN adopts a physically distributed and logically centralized multi-controller architecture. LLDP protocol is periodically used to obtain the link state information of the network, and a Merkle tree is establised according to the collected link information and the signature is generate based on KSI for each link that submitted by a switch by the main controller selected by using the PoW mechanism. Such, the dynamic change of network topology is recorded on Blockchian and the consistency of the topology information among multiple controllers can be guaranteed. The main controller issues the signature to the corresponding switch and a controller checks the legitimate of a switch by verifying the signature when it requests the flow rule table from the controller later. The signature verification ensures the authenticated communication between a controller and a switch. Finally, the simulation of the new scheme is implemented in Mininet platform that is a network emulation platform and experiments are done to verify our novel solution in our simulation tool. And we also informally analysis the security attributes that provided by our BCSDN.

Keywords: SDN, LLDP, Blockchain, KSI.

1. Introduction

With the rapid development of Internet application, the Software-Defined Networking (SDN) is coming into being to meet the increasing demand for the network traffics and have been widely applied in many areas [1-5]. While the SDN provides convenience in network management and operation, the split of the control plane and the data plane also maybe result in some security issues [3, 6-9], such as the single points of failure in the control plane, and the fragile channels between the control plane and the data plane, the data consistency in multiple controllers environment and all kinds of attacks initiated by a malicious switch and so on. The control plane is the most important component of

SDN. It is a key research area to resolve the problems that mentioned above. The redundant manner that uses the multi-controller architecture is a generic solution, such as the master-slave backup [10-13] and Byzantine Fault Tolerance scheme (BFT) [14-17]. In the master-slave backup architecture, these two controllers that are respectively called the master controller and the slave controller are used. When the master controller is shut down because of some failures or other reason, the slave controller is initiated and replaces the master controller to provide the network management task. The mechanism can ensure the data consistency and can improve the resiliency of the control layer. However, the method can't fundamentally solve the single point of failure. In [14-17], Byzantine fault-tolerant mechanisms based on Byzantine protocol are proposed to achieve the data consistency on different controllers and to confirm a failure controller. However, in these mechanisms, when the main controller fails, a view shift needs to be performed, which can cause a great overhead of network resources.

Blockchain [18-19] is a proof-tamper, and distributed database that is jointly maintained by multiple parties. It can achieve credible data sharing without the participation of a Trusted Third Party (TTP) and can increase the scalability and flexibility of the network. A survey about deployment of Blockchain in SDN is done in [20]. To solve the security problem of the control plane in SDN, some solutions based on Blockchain are proposed in [21-29]. However, these schemes still adopts the native structure of SDN about the internal structure of the controller, so they can't fully take advantage of the Blockchain features, and don't provide a systematic security mechanism.

In addition, the Link Layer Discovery Protocol (LLDP) [30] is a standard protocol of the network topology. However, some attacks such as the switch spoof, LLDP flood and so on, are found in [31]. Secondly, the establishment of a secure channel between the control plane and the data plane in SDN is also a hot research direction. The Transport Layer Security Protocol (TLS) [32] used by the Openflow protocol is by default a protocol between a controller and a switch. However, due to the complex configuration and the communication cost, TLS is considered as an alternative solution in later versions, which often lead to some security issues such as DDoS attack [33].

Keyless Signatures Infrastructure (KSI) [34] is a globally distributed system for providing timestamp and server-supported digital signature service. Only hash operation is used in KSI, so the scheme will not be impacted by some security problems such as a key leakage. KSI can ensure the long-term validity of digital signature and often is used for achieving a reliable communication. Using of KSI can prevent some attacks such as the switch spoof and so on. However, although KSI provides a complete cryptographic system, the core layer must calculate the root of the hash tree generated every time and publish it in the database. At present, there are still problems such as the lack of a credible mechanism for the database, the release cycle cannot meet the more fine-grained requirement, and the release channel must be a secure channel. The combination of Blockchain and KSI can solve these above problems. The hash calendar (Merkle tree) generated when the network topology change can be stored on Blockchain.

Aiming to the SDN security issues discussed before, a security framework for SDN based on Blockchain (BCSDN) is proposed in this paper. BCSDN adopts a physically distributed and logically centralized multi-controller architecture, and uses blockchain technology to build a unified database among controllers. All nodes in the network will periodically collect the link state information according to the instruction comes from

the main controller selected by the consensus scheme PoW [35-36]. A Merkle tree will be set up according to the link state information and the signature will be generated for each switch by the main controller based on KSI. The data consistency among multiple controllers can be ensured by using of the Blockchain. And the root hash value generated in every round will be written into the block header. The main controller will issue the signature to the corresponding switch. The proof-tamper, auditable and traceable features of Blockchain and KSI's features provide security guarantees. Finally, the simulation of BCSDN is implemented in Mininet platform [37] that is a network emulation platform and experiments are done to verify our novel solution in our simulation tool. And we informally analysis the security attributes that provided by our BCSDN.

The rest of the paper is organized as follows. Section 2 presents the related work. Some backgrounds on our solution are given in section 3, followed by our new scheme BCSDN in section 4. In section 5, security attributions of BCSDN are informally analyzed. Implementation and performance analysis of our solution are introduced in section 6. Finally we conclude our work in section 7.

2. Related Work

A redundant manner that uses the multi-controller architecture is a generic solution to resolve the single point of failure in SDN. The architecture is a scalable control plane solution for the large-scale SDN. To achieve high resilience, an SDN switch can connect one master controller for normal operation and one slave controller that backup the function of the master controller. Once the master controller fails, one of the slave controllers will be assigned to switches to works as the new master controller. However, the inappropriate slave controller assignment may cause controller chain failure, where running out of the capacity of the assigned controller, even crash the entire network. In [10], a dynamic slave controller assignment that prevents the network crash by planning slave controller assignment ahead of the controller failures is proposed. The controller chain failure phenomenon that incurred by unreasonable slave controller assignment can be solved. The slave controller assignment problem is formulated as a multi-objective mixed optimization problem that considers multiple network factors such as latency, load balancing and robustness. And it has been proven that it is a NP-complete complexity problem. A dynamic slave controller assignment (DSCA) scheme is introduced in [10]. DSCA firstly checks whether there are controller failures in state detection module, then completes the elastic slave assignment and generates a new slave assignment for switches in efficient slave assignment module. Finally, in role adjustment module, it changes the roles of some controllers and reconnects switches. Simulation results show DSCA can decrease the worst case latency under controller failures by 35.1% averagely, and reduce the probability of network crash.

In multi-controller architecture, the uneven distribution of traffic load in the controllers can degrade system performance. In [11], a self-adaptive load balancing (SALB) scheme that balances load among multiple controllers dynamically with multiple switch migration from source controllers to target controllers is proposed. The key feature of SALB is an effective distribution of load under high load condition while

considering the distance between switches and target controllers simultaneously. The efficacy of SALB is demonstrated through experimentation in [11] and the experimental results show that SALB experiences a small number of packet drops, which is less than 1.23% of the total number of message exchanges among the controllers.

Robustness and fault tolerance are two important metrics to be considered in assessing SDN's advantage. The currently available SDN controllers offer different fault tolerance mechanisms. In [12], existing fault-tolerant SDN controller solutions are surveyed and a mechanism is proposed to design a consistent and fault-tolerant Master-Slave SDN controller. The scheme [12] is able to balance consistency and performance. The main objective of [12] is to bring the performance of an SDN Master-Slave controller as close as possible to the one offered by a single controller. This is achieved by introducing a simple replication scheme, combined with a consistency check and a correction mechanism, that influence the performance only during the few intervals when it is needed, instead of being active during the entire operation time.

Despite many advantages of SDN, its deployment in the practical field is restricted since reliability and fault-tolerance capabilities of the system are not satisfactory. To overcome these difficulties of SDN, an architecture called FT-SDN has been proposed in [13]. The proposed architecture consists of a simple and effective distributed Control Plane with multiple controllers. FT-SDN uses a synchronized mechanism to periodically update the controller's state within themselves. In case of failure, FT-SDN has the ability to select another working controller based on the distance and delays among different network entities.

In the multi-controller architecture, most of state synchronization processes on different controllers depend on the assumption of a correct decision-making in the controllers. Successful introduction of SDN in the critical infrastructure networks also requires catering to the issue of unavailable, unreliable (e.g. buggy), and malicious controller failures. A framework tolerant to unavailability and Byzantine failures is proposed in [14]. It is called as MORPH. The MORPH can distinguish and localize faulty controller instances and appropriately reconfigure the control plane. A prototype SDN controller that can tolerate Byzantine faults in both the control and data planes is proposed in [15]. The performance of the novel solution is compared with current standard fault vulnerable open source SDN controllers. The experiment shows there is a reasonable slowdown of [15] as is expected in the transition from a fault vulnerable to a fault tolerant design. Their best controller can show only a 2x slowdown even though it only need 4 replica components, and so it can tolerate a single compromised component without affecting control and/or forwarding decisions in the networks. However, controllers in [15] are not fit for high performance levels to be adopted in large-scale networks.

A security framework based on the Byzantine protocol is proposed in [16]. In the scheme, controllers execute the Byzantine protocol and each switching device is managed by a controller view. The control information is given after multiple controllers arbitrate. By quantifying the heterogeneity of controllers, a two-stage controller view election algorithm is designed to ensure the availability of the network and the security of views.

Network survivability is the ability to maintain service continuity in the presence of failures. In [17], the network survivability of SDN is discussed in disaster situations. The solution in [17] considers multi-controller failure and the mechanism can reduce the

non-operational network devices in disaster situations. Preliminary results show that, by applying the proposed new approach, it is possible to achieve substantial improvements in network survivability.

To resolve security and privacy issues in SDN, some solutions based on Blockchain have explored. In [21], Blockchain Security over SDN (BSS) is proposed which protects privacy and availability of resources against non-trusting members. To verify their solution, mininet emulator is used for simulating custom SDN network topology. OpenDaylight controller is integrated with OpenStack controller. For testing purpose of Blockchain, Pyethereum tester tool under Ethereum platform is implemented. Serpent programming is used for creating contract in the blockchain. The simulation result shows that BSS facilitates files sharing among SDN users in distributed peer-to-peer basis using OpenStack as a cloud storage platform.

Since the large number of devices connected to the Internet of things (IoT) networks, the SDN-based network architecture makes the deployment and configuration of IoT much easier. In the IoT network, the fine-grained network traffic is critical to network management. In [22], a novel scheme based on Blockchain is proposed to measure the fine-grained network traffic in the SDN-based IoT networks and to ensure the security and consistency of the statistics. To measure flow traffic with low overhead and high accuracy, an ARIMA model and forecast the network traffic with the coarse-grained measurement of flows is designed. An objective function in ARIMA mode can decrease the estimation errors. A heuristic algorithm to obtain the optimal solution of the fine-grained measurement is used due to the objective function is an NP-hard problem.

To improve forwarding efficient of devices in the data plane of SDN, a method called TrustBlock is proposed in [23], which introduces trust as a security attribute in SDN routing planning. Besides, in order to enhance the integrity and controllability of trust evaluation, the double-layer blockchain architecture is established in [23]. In the first layer, the behavior data of the node is recorded, and then the trust calculation is performed in the second layer. In the evaluation model, nodes' trust is calculated from three aspects: direct trust, indirect trust and historical trust. Firstly, from the perspective of security, blockchain is used to achieve identity authentication of nodes, after that, from the perspective of reliability, the forwarding status is used to calculate the trust value. Secondly, consensus algorithm is used to filter malicious recommendation trust value and prevent colluding attacks. Finally, the adaptive historical trust weight is designed to prevent the periodic attack. In [23], the entropy method is used to determine the weight of each evaluation attribute, which can avoid the problem that the subjective judgment method is not adaptable to the weight setting. Simulation results show that the detection rate of the TrustBlock is up to 98.89%, which means this model can effectively identify the abnormal nodes in SDN. Moreover, it is attractive in terms of integrity and controllability.

In Software-Defined Networking (SDN), Northbound Interface provides APIs, which allow network applications to communicate with SDN controllers. However, a malicious application can access to SDN controller and perform illegal activities via these APIs. Although some studies proposed AAA (Authentication, Authorization, Accounting) systems to protect SDN controllers from malicious applications, their proposed systems also exist several limitations. Attackers can compromise a system, then modify its database or files to gain higher privileges. This system can be taken down because of Single Point of Failure threat. A novel system BlockAS is proposed to improve security

for the Northbound interface in [24]. It is used to authenticate, authorize and monitor accessing critical controller resources from applications. Specifically, BlockAS leverages Blockchain features to maintain the immutability and decentralization of credential data. In SDN, the lack of consistent records of network data poses difficulties for network management, and heterogeneous device heterogeneity poses a hindrance to software-defined network interoperability. [25] summarizes the development status and existing problems of software-defined network, proposes, realizes distributed consistent record of software-defined network data, and breaks the multi-vendor device isolation for fault recovery. Reduce the cost of network failure recovery and achieve unified scheduling of business capabilities. A security framework is also proposed that integrates Blockchain technology with multi-controller SDN in [26]. The main idea of the framework is to associate a set of controllers to each domain and to ensure a secure and trustworthy inter-controller communication. So, the proposed architecture considers a master controller and redundant controllers for each network domain. The architecture also integrates a reputation mechanism to identify a malicious controller. In [27], a distributed Blockchain-based SDN-IoT enabled architecture is proposed. It is the main goal of this framework to manage smart building. The traditional approach that manages the health-related data is often the centralized approach. It is not convenient to share and process electronic health data across the different institutions. In [28], an alternative way based on Blockchian technology is proposed to deal with information exchange across multiple stakeholders. A Blockchain-enabled Packet Parser (BPP) of the SDN is proposed in [29]. The scheme not only can detect attack in SDN and also can implement Blockchain protocol in data plane.

3. Research Background

3.1. The SDN Architecture

To resolve some issues in traditional network architecture, Software Defined Networks (SDN) is proposed. SDN is an emerging network architecture that decouples the control plane from the data plane and provides a software-based centralized controller. By this separation of control plane and data plane, switches in network become simple forwarding devices. Whereas, routing decision making is shifted to the controller, which can provide a global view of the network and a programming abstraction. This centralized entity provides a capability that an operator can program and real-time control underlying networks and devices. By using SDN, the network management becomes simply and helps in removing rigidity from the network.

The layered structure of SDN architecture, as shown in Fig. 1 has three major planes such as the data plane, the control plane, and the management plane. The data plane contains physical network elements, which form the path for data transmission. The control plane has a Network Operating System (NOS), also referred to as a controller, which generates the flow rule table for devices in data plane. These rules and policies are designed in the management plane of SDN architecture. The communication

between these planes is established by using well-defined Application Programmable Interfaces (APIs). These interfaces are divided into southbound, northbound, eastbound, and westbound APIs. The communications between the control plane and the data plane is implemented through the southbound API, which enables flow installation and configuration of devices. The control plane and the management plane use northbound API to provide programmability in SDN. Inter-controller communication of SDN domains is established using eastbound API, whereas westbound API is responsible for the legacy domain to SDN domain communication. The detail of these interfaces can be found in some literatures.

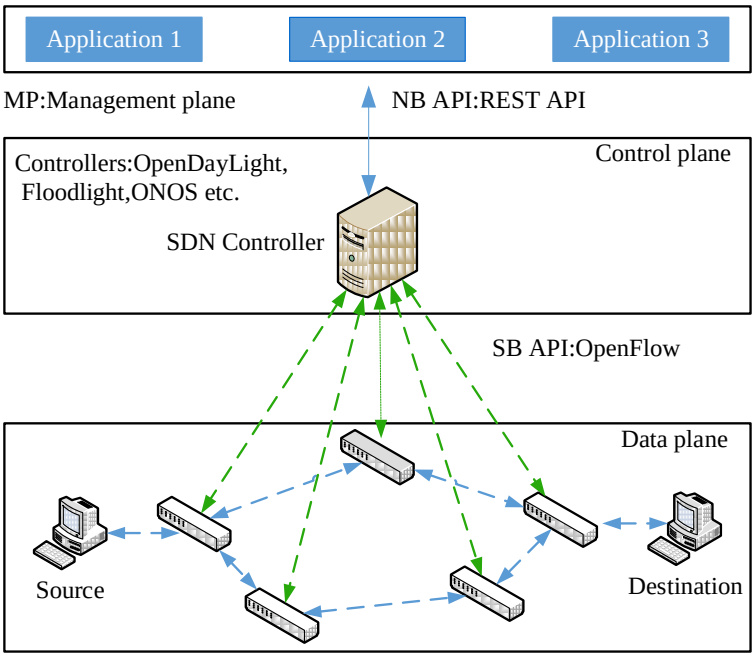


Fig. 1. The SDN Architecture [3]

3.2. The Link Layer Discovery Protocol

The decoupling between the control plane and the data plane introduced by SDN allows operators to employ remarkably cheap but very fast hardware to forward packets, moving the control logic to the much smarter controller. The controller plays the role of an operating system of the network. One of fundamental functions that a controller must offer is an accurate, nearly real time view of the network topology. This function is known as the topology discovery. The Link Layer Discovery Protocol (LLDP) [30] is a standard method of the network topology discovery in SDN. Fig. 2 shows the principle that how LLDP works. To discover the unidirectional link $s_1 \rightarrow s_2$, the controller

encapsulates a LLDP packet in a Packet-out message and sends it to s_1. The Packet-out contains instruction for s_1 to send the LLDP packet to s_2 via port p_1. When s_2 receives the LLDP packet via port p_2, s_2 encapsulates it as a destination switch in a Packet-in message and sends it back to the controller. The controller receives the LLDP packet and concludes that there is a unidirectional link from s_1 to s_2. The same process is performed to discover the opposite direction $s_2 \rightarrow s_1$ as well as all other links in the network. After all switches perform such operations, the controller will obtain the network topology information of the entire network. However, the network topology will dynamically change incurred by switches leave and join the network. So the controller needs to periodically repeat the process described in Fig. 2.

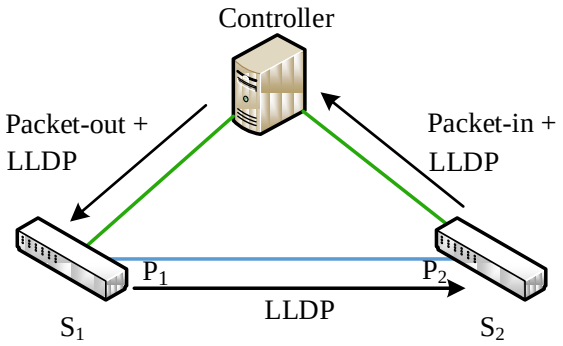


Fig. 2. The link discovery process [30]

3.3. Blockchain and PoW

Blockchain technology has been applied in many areas [18-19]. Blockchain is a system that is composed of nodes, communicating with each other through a protocol. A node can be a physical machine or a virtual machine. The IP address is used to identify the node in the Blockchain network. The public key is used as an user identification in the network. The private key is generally used for signing on message transmitted on the network. As a result, each user can log in from any node in the system. The consistency of data stored on Blockchain must be guaranteed on the entire network and can be achieved by some consensus algorithm such as PoW, PoS and so on [36]. And data on the Blockchain network is digitally signed to guarantee authenticity and accuracy properties. Blockchain technology can ensure an immutable storage and a fraud protection property. The work mechanism of the Blockchain network is shown in Fig. 3. The transaction data is stored in a specific data structure called “block” in Blockchain. The blocks generated during the transaction process are linked together via the cryptographic hash function to form a chain of blocks. That is to say, each block inside the Blockchain stores a hash value of the previous block. Thus, the chain of blocks is grouped or linked in a chronological order. As a result, the data that stored on the

Blockchain won't be modified without cooperation of all nodes inside the system. So, the mechanism provides a proof-tamper feature.

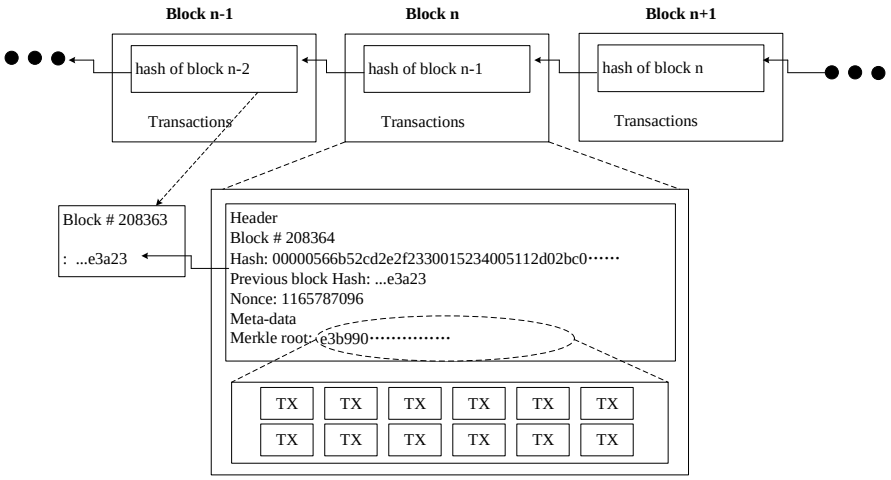


Fig. 3. The Blockchain and block structure [18]

Blockchain technology has been applied in many areas [18-19]. Blockchain is a system that is composed of nodes, communicating with each other through a protocol. A node can be a physical machine or a virtual machine. The IP address is used to identify the node in the Blockchain network. The public key is used as an user identification in the network. The private key is generally used for signing on message transmitted on the network. As a result, each user can log in from any node in the system. The consistency of data stored on Blockchain must be guaranteed on the entire network and can be achieved by some consensus algorithm such as PoW, PoS and so on [36]. And data on the Blockchain network is digitally signed to guarantee authenticity and accuracy properties. Blockchain technology can ensure an immutable storage and a fraud protection property. The work mechanism of the Blockchain network is shown in Fig. 3. The transaction data is stored in a specific data structure called "block" in Blockchain. The blocks generated during the transaction process are linked together via the cryptographic hash function to form a chain of blocks. That is to say, each block inside the Blockchain stores a hash value of the previous block. Thus, the chain of blocks is grouped or linked in a chronological order. As a result, the data that stored on the Blockchain won't be modified without cooperation of all nodes inside the system. So, the mechanism provides a proof-tamper feature.

3.4. Proof of Work

Blockchain is a key technology to build a distributed trust in the environment that users don't trust each other and there doesn't exist a Trust Third Party (TTP). In Blockchain network, the consensus scheme ensures the consistency of data stored on the Blockchain. More recently, some consensus schemes for Blockchain have been

proposed and most of them are based on three basic algorithms that often used in a distributed network, such as Proof of Work (PoW) [35], Proof of Stake (PoS) and Direct Acyclic Graph (DAG). A comprehensive performance comparison is done among them in [36].

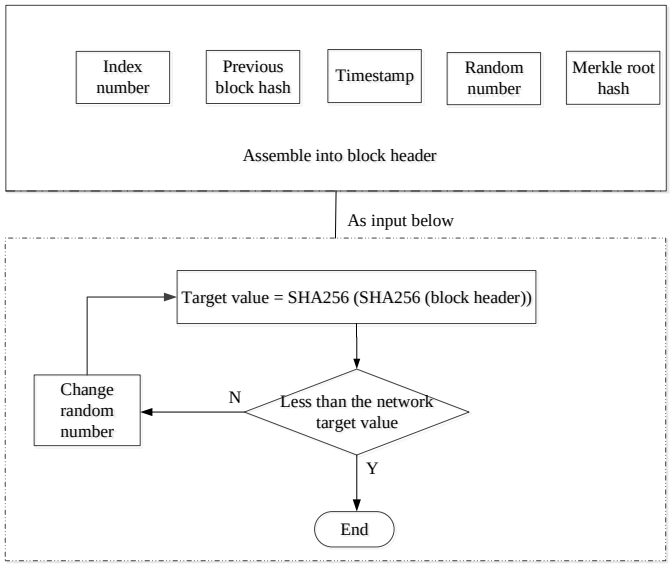


Fig. 4. Proof of Work

PoW [35] used in Bitcoin is the most classical consensus algorithm in the Blockchain. The PoW involves a scanning of a hash value that computed by using a hash algorithm such as SHA-256. The hash value begins with a string of 0 bits. The average workload is exponent in the number of 0 bits required and can be verified by executing a single hash. The PoW is implemented by incrementing a nonce in the block until a hash value that contains the required number of 0 bits in the block's hash. Fig. 4 shows how the PoW works. Once the computed value satisfies the requirement of PoW, the block cannot be changed without re-executing the work. As subsequent blocks are chained to the new generated block, modifying a block means regenerating all the blocks after the modified block. So the core idea of PoW used in Blockchain is that miners use their computing power to compete the hashing operation. The winner who first finds the hash value lower than the announced target has the right to insert a new block into the blockchain and get a certain amount of reward.

3.5. Keyless Signature Instructure

The Keyless Signature Instructure (KSI) [34] is a globally distributed system for providing digital signature services. KSI is an alternative solution to traditional PKI signature. It has some benefits and has been payed widely attention. It can detect the change status of digital assets and submit this information for further audit and

investigation. A mechanism with multiple signatures can be obtained in KSI. That is to say, multiple documents can be signed together at once. The signing process includes the following three steps. *Hash*: a hash value of the data or file generated by the client will be calculated; *Aggregation*: The gateway layer collects and processes the hash values that comes from the clients, aggregates them into a Merkle tree, and sends the generated root hash value to the aggregation layer. The aggregation layer server processes the root hash value generated and sent by the gateway layer, and adds it to the Merkle tree. Finally, the generated root hash value will be transferred to the core layer; *Release*: a permanent hash tree will be created according to the first three hash values of the aggregation tree collected each time and it is released as a trust anchor.

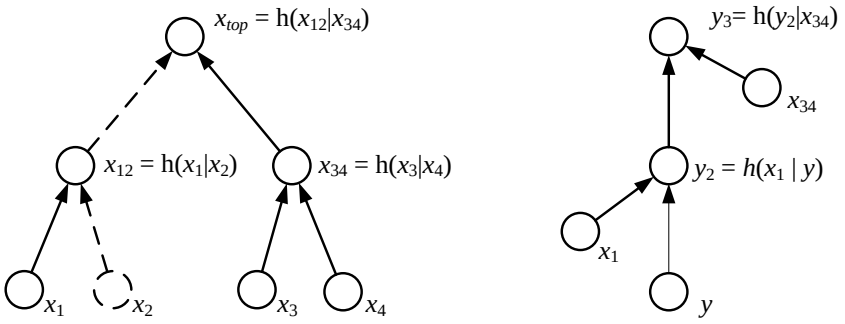


Fig. 5. Hash Tree and Hash Calendar. [34]

Hash Trees: Hash-tree aggregation process described before was first introduced in [38]. In hash-tree time-stamping scheme, a one-way hash function is used to convert a list of data or files into a fixed length hash value that is generally associated with time. A signature token generated by the service according to a hash of a document from client is considered as a proof that the data sent by the client existed at the given time and that the request was received through a specific access point. All received requests are aggregated together into a large hash tree; and the top of the tree is fixed and retained for each second as shown in Fig.5. The signature token contains data for reconstructing a path through the hash tree—starting from a signed hash value (a leaf) to the top hash value. For example, to verify a token y in the place of x_2 (Fig. 5), a concatenation operation is firstly done between y and x_1 (retained as a part of the signature token) and then a hash value $y_2 = h(x_1 \mid y)$ is calculated and is used as the input of the next hash step, the process will be end when it reach the top hash value, i.e. $y_3 = h(y_2|x_{34})$ in the example case. If $y_3 = x_{top}$ then it is safe to prove that y was in the original hash tree.

Hash Calendar: These top hash values obtained in each round are linked together to generate a globally unique hash tree (The hash tree is called a hash calendar in [34])—so that new leaves are added only to one side of the tree. Time value is encoded as the shape of the calendar—the modification of which would be evident to other users. However, the top hash of the calendar is required to periodically publish in widely witnessed media. There is a deterministic algorithm to compute the top hash of the linking hash tree, giving a distinct top level hash value at each second. Also there is an algorithm to extract time value from the shape of the linking hash tree for each second, giving a hard-to-modify time value for each issued token.

4. BCSDN Framework

The BCSDN architecture proposed in this paper is a distributed multi-controller architecture. In the control plane of BCSDN, the controllers collect the link status information from each switch that joined to the network, by using the link discovery protocol LLDP during the link discovery phase. A switch will package the link information in a Packet_in packet during the link discovery, and then it will submit the Packet_in packet to a controller. In BCSDN, the submission is considered as a transaction process in the Blockchain network. The consensus algorithm such as PoW is used to elect a main controller from these controllers. The selected main controller that plays a role of a miner verifies the transaction and aggregates all of hash values to generate a Merkle tree according to KSI algorithm. Finally, it will generate a block of the root hash value and storage on the Blockchian network, and then it will issue signature to each switch according to KSI signature rule. Each block is related to a hash calendar. Thus, the chain of blocks records and represents the dynamical change process of the network topology. The main controller will issue the latest network topology information collected from the network to other controllers so that the scheme ensures the consistency of the network view among controllers. When a switch in the network needs to forward a data, interaction will be performed between the switch and the controller that directly connected with the switch to request the flow rule table. The controller sends the latest flow rule table after verifying the signature owned by the switch. So, our BCSDN framework is shown in Fig. 6 and it is consisted of the following 5 components: The network topology generation, Blockchain establishment, the selection of the main controller, the signature generation and signature verification.

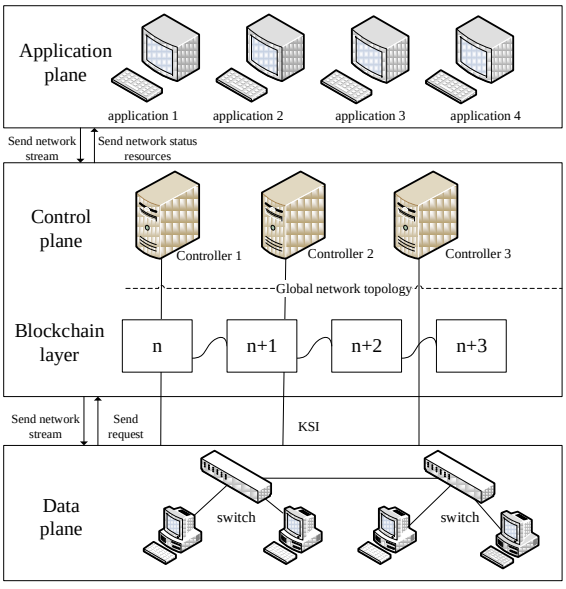


Fig. 6. The Entire Framework of BCSDN

4.1. The Network Topology Generation

In BCSDN, the standard link layer discovery protocol LLDP introduced in section 3.2 is used to collect the network link state information. The SDN controller initiates the link discovery process. The process consists of the following 4 steps. (1) The SDN controller periodically sends a LLDP packet Packet_out packet to all switches that connected with it. (2) Once a switch receives the Packet_Out packet from the SDN controller, it will broadcast the Packet_out packet to all of devices that connected with the switch via all of its ports. (3) In our BCSDN, we assume that the neighboring switches are an OpenFlow switch. That is to say, the switch have no a special flow rule entry for processing LLDP messages, so they will send a LLDP packet Packet_in packet to the controller connected with them. (4) After the controller receives a Packet-In packet, it will analyze the data packet and save the link information between the two switches in its link discovery table and calculate a hash value of message in the Packet-In packet by using SHA256. The algorithm is described in Algorithm 1.

Algorithm1: Hash the Link Information

Input: Packet_out{}
Output: temptxList{}
1: while a switch receives packet_out{} message do
2: forward the message to neighboring switches
3: neighboring switches sends the packet_in to the controller
4: temptxList{} ← sha256(packet_in)
5: return temptxList{}

4.2. Blockchain Establishment

Algorithm2: Merkle Tree Construction

Input: temptxList{}
Output: root
1: while newTxList.size() != 1 do
2: index = 0
3: while index < tempTxList.size() do
4: left ← tempTxList.(index)
5: index++
6: right ← " "
7: if index != tempTxList.size() then
8: right = tempTxList(index)
9: newTxList{} ← SHA256(left , right)
10: index++
11: root ← newTxList(0)
12: return root

In BCSDN, the main controller will generate a Merkle tree according to the KSI scheme during the link discovery process. The main controller will save the hash value of the root node as a block on the Blockchain maintained by these controllers according to the

principle of Blockchain. A new block on Blockchain is set up as shown in Fig. 6. That is to say, an update (a hash calendar) of the network topology will generate a new block on the Blockchain. That is to say, the Merkle tree locally represents the current network topology information. The construction algorithm of the Merkle tree is shown in Algorithm 2.

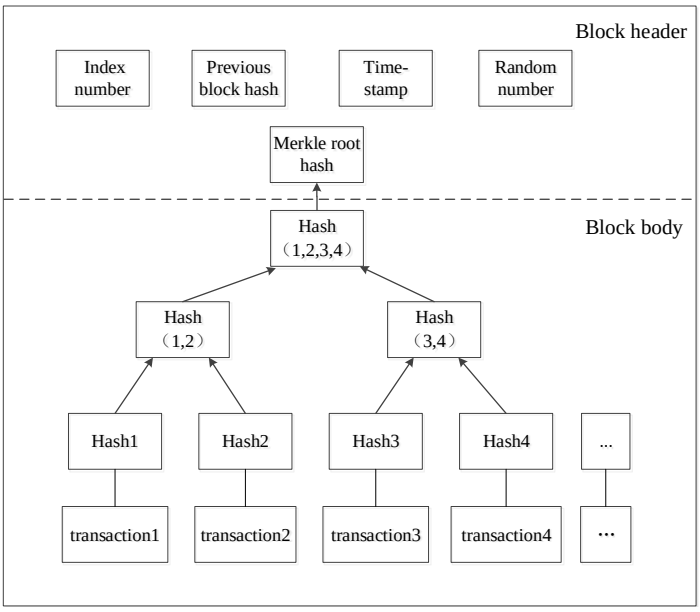


Fig. 6. A block generation on Blockchain

4.3. Selection of a Main Controller

In order to solve the single-point failure problem incurred by a single controller in SDN, a multi-controller architecture is adopted in our BCSDN framework. These controllers are deployed in physically distributed and logically centralized manner. The PoW introduced in section 3.4 is used for selecting a main controller from these controllers. The algorithm is described in Algorithm 3.

The main controller just selected will play a role of a miner in the Blockchain network and issue the topology information to all of controllers. So, our novel BCSDN can ensure the consistency of the network topology information on these different controllers.

Algorithm3: Proof of Work

Input: index, rootHex, time, previousBlockHash, random, TargetValue
Output: Block
1: Block ← index + rootHex + time + previousBlockHash + random
2: while true do
3: if SHA256(SHA256(Block)) >= TargetValue then
4: random ++
5: Block ← index + rootHex + time + previousBlockHash + random
6: else
7: break
8: end if
9: end while
10: return Block

4.4. The Signature Generation

The main controller selected in PoW process will manage the entire network. The main controller will use the KSI scheme to generate and issue a signature to each switch that submitted the correct link information to it. The signature information generated by the main controller is a concatenation of these hash values on nodes which is located on the Merkle tree. These nodes together form a path from a leaf node that represents information of a link to the root of the Merkle tree. So, this signature means that the switch sent a Packet_In packet containing the link information to construct the Merkle tree that recorded the current network topology information. The signature process is shown in Algorithm 4.

Algorithm4: Signature Generation

Input: CalculationPath, node
Output: HashSignature{}
Initialization: HashSignature{}= ∅
1: if the node is in CalculationPath and is a leaf
2: HashSignature{} ← node
3: if the node is in CalculationPath
4: traversing left and then right
5: otherwise HashSignature{} ← node
6: return HashSignature{}

4.5. Signature Verification

When a switch need to forward data for an end user, it firstly checks if there is a matching entry in a flow rule table. If the check fails, a forwarding request event is generated and then the switch sends the request with signature the main controller issued to it to the controller connected with it. When the controller receives the request, it verifies the signature of the switch according to KSI algorithm. If the verification is successful, the controller transfers the latest flow rule table to the switch. Otherwise, it drops the request and don't response to the switch. The algorithm of signature verification is described in Algorithm 5. The signature scheme used in BCSDN can

efficiently authenticate the switch and prevent some attacks such as spoofing incurred by malicious switches controlled by adversary.

Algorithm5: Signature Verification

Input：HashSignature{node 0,node 1…node i}, root
Output：result
Initialization: HashNode = ∅
1： HashNode ← node 0
2： for (n=1, n<=i, n++)
3： HashNode= sha256(HashNode, node n)
4： if HashNode == root then
5： result ← verify successfully
6： break
7： else
8： result ← verify failed
9： end if
10：return result

5. Security Analysis

In this section, we informally discuss the security issues solved in our BCSDN. BCSDN can efficiently solve the single point of failure, the view consistency of multi-controller SDN network and the authentication of the interaction between a controller and a switch.

***Proposition* 1**: BCSDN can solve the single point of failure.

Proof: In BCSDN, the multi-controller architecture is used to solve the single point of failure. That is to say, a logically centralized and physically distributed multi-controller framework is adopted in the control plane of SDN. The selected main controller manages the entire network. When the main controller shut down because of some reason, the re-elected main controller will take over the network management task. Therefore, the multi-controller architecture can conquer the single point of failure, improve the processing capacity of the control plane, and also ensure the reliability of the network management.

***Proposition* 2**: BCSDN can ensure the consistency of the network topology on the different controllers.

Proof: In BCSDN, the mechanism based on Blockchain is implemented in the control plane. The dynamic change of the network topology will be recorded in Blockchain. These features of Blockchain such as proof-tamper, auditable, distributed storage and so on will ensure that the network topology information stored in the Blockchain is correct and won't be modified and also guarantee that the network topology information stored on the different controllers is consistent.

***Proposition* 3**: The communication between a controller and a switch can be authenticated.

Proof: In BCSDN, KSI is used when a controller collect the topology information of the network. That is to say, the controller will generate a signature for each switch according to the KSI mechanism in link discovery process. The signature is a proof that

proves the switch ever took part in the link discovery process in the corresponding network topology. That is to say, each link (that submitted by the switch) in the network topology is a legal link. The KSI scheme ensures that the signature that the controller sent to each switch can't be forged and these signatures information are saved on Blockchain. When a switch that needs to forward data requests the flow rule table from a controller, the controller will verify the signature generated in link discovery process and owned by the switch. So, using of KSI can ensure the authenticated communication between a controller and a switch.

6. BCSDN Implementation

6.1. BCSDM simulation Implementation

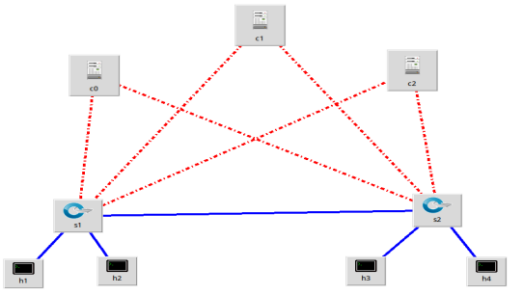


Fig. 7. The Simulated Network Topology.

In this section, we implement our BCSDN architecture by using simulation method. The network emulator Mininet on Ubuntu 16.04 system is used to simulate a custom topology of a SDN network. The Floodlight controller is used to establish a multi-controller architecture for SDN to manage the entire network. We illustrate the operation of the BCSDN network by a simply network topology as shown in Figure 7. Three Floodlight controllers are used to establish a multi-controller architecture. These controllers together form a simply p2p network and are used to manage and maintain a Blockchain network. The main controller is selected by using the PoW alogrithm. We implement the network architecture instance of BCSDN in the network simulation platform Mininet. These controllers in BCSDN periodically collect the network link information by the link discovery protocol LLDP introduced in 3.2 section. The main controller generates the signature corresponding to each link according the Merkle tree that presents the current network topology. The network state information is recorded on the Blockchain as a distributed ledger. The simulation topology includes 2 OpenFlow switches and 4 hosts connected to these two switches respectively. Built-in Mininet network commands such as "ping" and so on can be used to test the correctness,

feasibility and overhead of BCSDN. The details of each node in the simulation topology are described as shown in Figure 8. In Fig. 9a, the information from the log file of the node displays the block creation and the collection of the network topology. In Fig. 9b, the signature verification result is shown when the switch applies for the flow rule table from the controller.

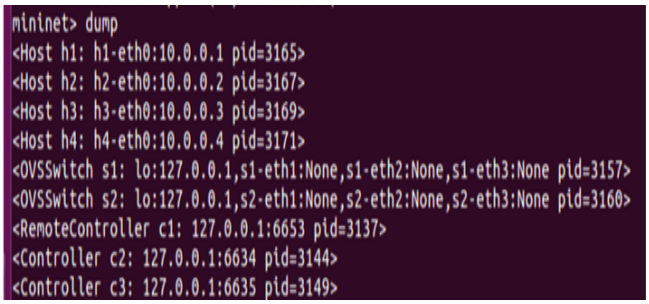


Fig. 8. Information of the Network Nodes.

2020-08-11 09:04:21.988 INFO [n.f.l.i.LinkDiscoveryManager]
Received LLDP packet on sw 00:00:00:00:00:00:00:01, port 2

The first block: Block{
index=0,
rootHex='IFmBkeOfebmNHm0qTf508QsMrfPTflqMt5yKph8LsYU=',
time=1597161862025,
previousBlockHash=0,
random=945}

The second block: Block{
index=1,
rootHex='65Bm8Zbb69XU3pWHDusPh8mc7sicAm9TmRqJoFlwcTs=',
time=1597161862064,
previousBlockHash=NebGApj7T6pIv7Ho87IbMNDdfPLOrE3ix1wU7D0mjm4=,
random=388}

a. Block information

2020-08-11 09:05:46.865 INFO [n.f.f.Forwarding] sign pass

b. The Signature Verification

Fig. 9. The results of the Network Simulation.

6.2. Performance Analysis

The performance of BCSDN is analyzed under different network scale in this section. We compare performance metrics such as the network convergence time, network

throughput and the response time between the singe controller solution and our BCSDN. The Floodlight controller is used in the experimental analysis.

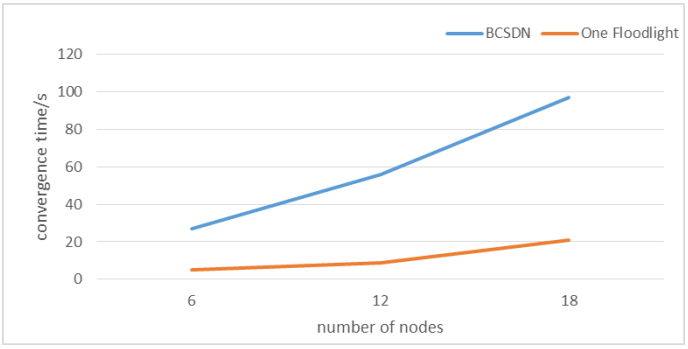


Fig. 10. The Network Convergence time

Fig.10 shows the convergence time of these two simulated networks that one only uses a single controller and another uses our BCSDN solution. It can be clearly seen that these different networks can converge successfully in three different network sizes and obtain the entire network topology. BCSDN needs to set up Blockchain network according to the change of the network topology. So, the convergence time is higher than the solution used the single controller. But, we can see that the curve of the convergence time in BCSDN will become smooth as the network size increases. That is to say, BCSDN is better than the network that there exists only a single controller in large scale network.

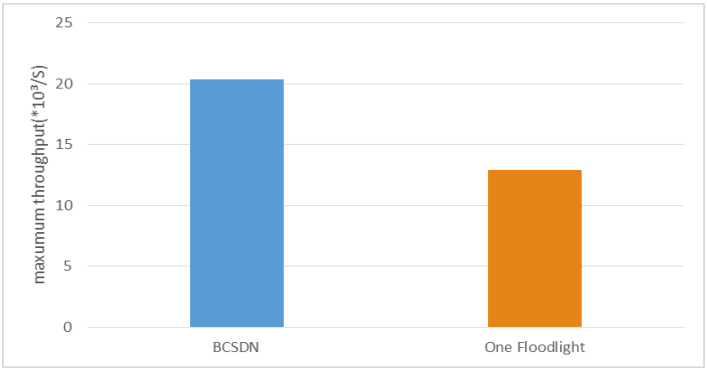
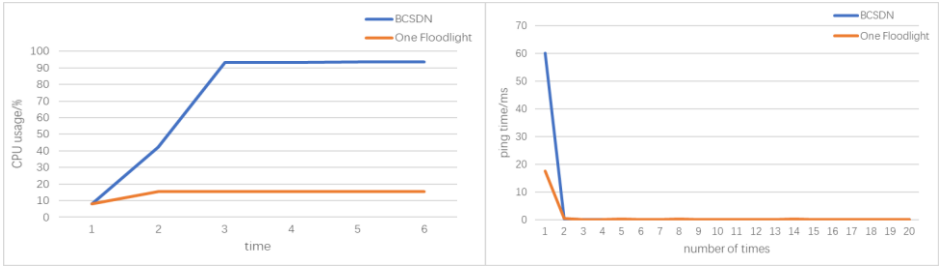


Fig. 11. Throughput Comparison Between the Sing Controller network and BCSDN

The network throughput is compared between BCSDN and the network with the single controller as shown in Figure 11. In BCSDN, three controllers are used. the processing ability is indeed stronger than the network with a single controller. This shows that our multi-controller network architecture is more excellent than the single controller network architecture.



a. CPU usage b. The Response Time

Fig. 12. Comparison of CPU usage and the Response Time

In addition, the CPU usage rate and the response time are analyzed in these two solutions as shown in Fig.12. Clearly, because using of Blockchian, the CPU usage rate in BCSDN is higher than the solution that used a single controller as Fig. 12a. However, the response time is tested by using performing 20 ping operations. As shown in Figure 12b, it can be seen that the ping response time of BCSDN and a single controller network is almost the same except that the time of the first ping. That is to say, BCSDN can meet the requirements of network rapid processing.

7. Summary

In this paper, a security framework for SDN BCSDN is proposed by integrating Blockchain and KSI in this paper. The BCSDN adopts a physically distributed and logically centralized multi-controller architecture. In BCSDN, LLDP protocol is used to obtain the topology information of the network, and the dynamic change of the network topology is recorded on Blockchain. The main controller selected according to PoW play a role that manage the entire network. So, BCSDN can efficiently solve the single point of failure in single-controller architecture. In addition, using of Blockchain scheme ensures the consistency of the topology information on different controllers. Using of KSI is used to authenticate the communication between a controller and a switch. The correctness, reliability and feasibility are verified by an emulation method in mininet emulation platform in this paper. We also simply analyze security attributes and performance of the BCSDN framework. We will further improve our solution in the future work.

Acknowledgments. This work is supported by NSFC No. 61461027; Gansu province science and technology plan project under grant No. 20JR5RA467; Innovation Promotion Education Fund of Ministry of Education No. 2018A05003.

References

1. Rishikesh Sahay, Weizhi Meng, Christian D. Jensen.: The application of Software Defined Networking on securing computer networks: A survey [J]. Journal of Network and Computer Applications, Vol. 131, 89-108. (2019)
2. Y. Zhao, Y. Li, X. Zhang, G. Geng, W. Zhang and Y. Sun.: A Survey of Networking Applications Applying the Software Defined Networking Concept Based on Machine Learning [J], IEEE Access, Vol. 7, 95397-95417. (2019)
3. Ali, J.; Lee, G.-M.; Roh, B.-H.; Ryu, D.K.; Park, G.: Software-Defined Networking Approaches for Link Failure Recovery: A Survey [J], Sustainability, Vol. 12, No. 10, 4255. (2020)
4. Sanjeev Singh, Rakesh Kumar Jha.: A Survey on Software Defined Networking: Architecture for Next Generation Network [J]. Journal of Network and Systems Management, Vol. 25, 321-374. (2016)
5. Hu T, Guo Z, Yi P, et al. Multi-controller based software-defined networking: A survey [J]. IEEE Access, Vol. 6, 15980-15996. (2018)
6. Heng Zhang, Zhiping Cai, Qiang Liu, Qingjun Xiao, Yangyang Li, Chak Fone Cheang.: A Survey on Security-Aware Measurement in SDN [J], Security and Communication Networks, vol. 2018, 14 pages. (2018)
7. Y. Liu, B. Zhao, P. Zhao, P. Fan and H. Liu.: A survey: Typical Security Issues of Software-Defined Networking [J], China Communications, vol. 16, no. 7, 13-31. (2019)
8. Juan Camilo Correa Chica, Jenny Cuatindioy Imbachi, Juan Felipe Botero Vega.: Security in SDN: A Comprehensive Survey [J], Journal of Network and Computer Applications, Vol. 159, 102595. (2020)
9. Tao Han, Syed Rooh Ullah Jan, Zhiyuan Tan, et. al.: A Comprehensive Survey of Security Threats and Their Mitigation Techniques for Next-generation SDN Controllers [J], Concurrency and Computation：Practice and Experience, Vol. 32. (2020)
10. Tao Hu, Peng Yi, Zehua Guo, Julong Lan, Yuxiang Hu.: Dynamic slave controller assignment for enhancing control plane robustness in software-defined networks [J], Future Generation Computer Systems, Vol. 95, 681-693. (2019)
11. Madhukrishna Priyadarsini, Joy Chandra Mukherjee, Padmalochan Bera, Shailesh Kumar, A. H. M. Jakaria, M. Ashiqur Rahman. An adaptive load balancing scheme for software-defined network controllers, Computer Networks, Vol. 164 (2019)
12. A. J. Gonzalez, G. Nencioni, B. E. Helvik and A. Kamisinski.: A Fault-Tolerant and Consistent SDN Controller [C], In Proceeding of 2016 IEEE Global Communications Conference (GLOBECOM), Washington, DC, 1-6 (2016)
13. Das, R.K., Pohrmen, F.H., Maji, A.K. et al.: FT-SDN: A Fault-Tolerant Distributed Architecture for Software Defined Network [J]. Wireless Personal Communication, Vol. 114, 1045–1066. (2020).
14. E. Sakic, N. Đerić and W. Kellerer. : MORPH: An Adaptive Framework for Efficient and Byzantine Fault-Tolerant SDN Control Plane [J], IEEE Journal on Selected Areas in Communications, Vol. 36, No. 10, 2158-2174. (2018)
15. K. ElDefrawy and T. Kaczmarek. : Byzantine Fault Tolerant Software-Defined Networking (SDN) Controllers [C], In Proceeding of 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC), Atlanta, GA, 208-213. (2016)
16. Gao J., Wu J. X.,Hu Y. X,et al. Research on Anti-attack of Software-Defined Network Control Surface Based on Byzantine Fault Tolerance [J]. journal of Computer Applications, Vol. 37, No. 8, 2281-2286. (2017) (in Chinese).
17. Luis Guillen, Hiroyuki Takahira, Satoru Izumi, Toru Abe, Takuo Suganuma.: On Designing a Resilient SDN C/M-Plane for Multi-Controller Failure in Disaster Situations [J], IEEE Access, Vol. 8, 141719-141732. (2020)

18. Dharmin Dave, Shalin Parikh, Reema Patel, Nishant Doshi.:A Survey on Blockchain Technology and its Proposed Solutions [J], Procedia Computer Science, Vol. 160, 740-745. (2019)
19. Gamage, H.T.M., Weerasinghe, H.D. & Dias, N.G.J.: A Survey on Blockchain Technology Concepts, Applications, and Issues [J]. SN Computer Science, Vol. 1, 114. (2020).
20. TALAL ALHARBI.: Deployment of Blockchain Technology in Software Defined Networks: A Survey [J], IEEE Access, Vol. 8, 9146-9156. (2020)
21. S. R. Basnet and S. Shakya.: BSS: Blockchain security over software defined network [C], In Proceeding of the 2017 International Conference on Computing, Communication and Automation (ICCCA), Greater Noida, 720-725. (2017)
22. Huo, L., Jiang, D., Qi, S. et al.: A Blockchain-Based Security Traffic Measurement Approach to Software Defined Networking [J]. Mobile Networks and Applications, (2020)
23. Bo Zhao, Yifan Liu, Xiang Li, Jiayue Li, Jianwen Zou.: TrustBlock: An adaptive trust evaluation of SDN network nodes based on double-layer blockchain [J], PLoS One, Vol. 15, No. 3, e0228844. (2019).
24. Hien Do Hoang, Phan The Duy, Van Hau Pham.: A Security-Enhanced Monitoring System for Northbound Interface in SDN using Blockchain [C], In Proceedings of the Tenth International Symposium on Information and Communication TechnologyDecember, NY, USA, 197–204. (2019)
25. C. Xue, N. Xu and Y. Bo. : Research on Key Technologies of Software-Defined Network Based on Blockchain [C], In Proceeding of 2019 IEEE International Conference on Service-Oriented System Engineering (SOSE), San Francisco East Bay, CA, USA, 239-2394. (2019)
26. A. Derhab, M. Guerroumi, M. Belaoued, O. Cheikhrouhou. : BMC-SDN: Blockchain-Based Multicontroller Architecture for Secure Software-Defined Networks, Wireless Communications and Mobile Computing, vol. 2021, Article ID 9984666, 12 pages. (2021)
27. A. Rahman, M. K. Nasir, Z. Rahman, A. Mosavi, S. S. and B. Minaei-Bidgoli. DistBlockBuilding: A Distributed Blockchain-Based SDN-IoT Network for Smart Building Management [J], IEEE Access, vol. 8, 140008-140018. (2020)
28. D. Zlate, F. Sonja, M. Anastas, T. Vladimir. : Real time availability and consistency of health-related information across multiple stakeholders: A blockchain based approach [J], Computer Science and Information Systems, Vol. 18, No. 3, 927-955. (2021)
29. Abbas Yazdinejad, Reza M. Parizi, Ali Dehghantanha, Kim-Kwang Raymond Choo.: P4-to-blockchain: A secure blockchain-enabled packet parser for software defined networking [J], Computers & Security, Vol. 88. (2020)
30. W.-Y. Huang, T.-Y. Chou, J.-W. Hu, and T.-L. Liu. : Automatical end to end topology discovery and flow viewer on SDN [C], In Proceeding 2014 28th International Conference on Advanced Information Networking and Applications Workshops, Victoria, BC, 910-915. (2014)
31. Mowla Nishat I, Doh Inshil, Chae Kijoon,: CSDSM: Cognitive switch-based DDoS sensing and mitigation in SDN-driven CDNi word [J], Computer Science and Information Systems, Vol. 15, No. 1, 163-185. (2018)
32. Dierks T, Rescorla E. The transport layer security (TLS) protocol version 1.2. RFC 5246, 1-104. (2008).
33. A. Azzouni, N. T. Mai Trang, R. Boutaba and G. Pujolle. : Limitations of openflow topology discovery protocol [C], In Proceeding of 2017 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), Budva, 1-3. (2017)
34. Buldas A., Kroonmaa A., Laanoja R.: Keyless Signatures' Infrastructure: How to Build Global Distributed Hash-Trees [C], Nordic Conference on Secure IT Systems, Lecture Notes in Computer Science, Vol. 8208, 313-320.(2013)
35. Bin Cao, Zhenghui Zhang, Daquan Feng, et. al. : Performance analysis and comparison of PoW, PoS and DAG based blockchains [J], Digital Communications and Networks, Vol. 6, 480-485. (2020)

36. Satoshi Nakamoto.: Bitcoin: A Peer-to-Peer Electronic Cash System, https://bitcoin.org/en/bitcoin-paper.
37. R. L. S. de Oliveira, C. M. Schweitzer, A. A. Shinoda and Ligia Rodrigues Prete. : Using Mininet for emulation and prototyping Software-Defined Networks [C], In Proceeding of the 2014 IEEE Colombian Conference on Communications and Computing (COLCOM), Bogota, 1-6. (2014)
38. Merkle, R.C.: Protocols for public-key cryptosystems [C]. In Proceedings of the 1980 IEEE Symposium on Security and Privacy, Oakland, CA, USA, 122–134 (1980)

Xian Guo, is an associate professor of School of Computer and Communication, Lanzhou University of Technology. He is a visiting scholar at University of Memphis. He received MS and PhD in Lanzhou University of Technology, China, in 2008 and 2011, respectively, and BS in Nothwest Normal University. His current research interests include network and information security, cryptographic, and blockchain. E-mail: iamxg@ 163.com.

Chen Wang, is currently a master student at Computer and Communication School of Lanzhou University of Technology. He received his Bachelor degree from Lanzhou University of Technology in 2017, and started his master studying in 2018. His research interests are Software Defined Networking, security of wireless network, blockchain.

Laicheng Cao, is a professor of School of Computer and Communication, Lanzhou University of Technology. He received MS in Lanzhou University, China, in 2004. His current research interests include network and information security, cryptography etc.

Yongbo Jiang, is a lecturer of School of Computer and Communication, Lanzhou University of Technology. He received MS and PhD in Xidian University, China, in 2008 and 2013, respectively. His current research interests include network and information security, information-centric networking etc.

Yan Yan, is an associate professor of School of Computer and Communication, Lanzhou University of Technology. She received MS and PhD in Lanzhou University of Technology, China, in 2005 and 2018, respectively. Her current research interests include machine learning, privacy computing, and blockchain.

Received: February 02, 2021; Accepted: December 27, 2021.

Computer Science and Information Systems 19(2):547–572 https://doi.org/10.2298/CSIS210217061E

Reasoning on the Usage Control Security policies over Data Artifact Business Process Models

Montserrat Estañol[1], Ángel Jesús Varela-Vaca[2], María Teresa Gómez-López[2], Ernest Teniente[1], and Rafael M. Gasca[2]

[1] Universitat Politècnica de Catalunya
Barcelona, Spain
{estanyol,teniente}@essi.upc.edu
[2] Universidad de Sevilla
Sevilla, Spain
{ajvarela,maytegomez,gasca}@us.es

Abstract. The inclusion of security aspects in organizations is a crucial aspect to ensure compliance with both internal and external regulations. Business process models are a well-known mechanism to describe and automate the activities of the organizations, which should include security policies to ensure the correct performance of the daily activities. Frequently, these security policies involve complex data which cannot be represented using the standard Business Process Model Notation (BPMN). In this paper, we propose the enrichment of the BPMN with a UML class diagram to describe the data model, that is also combined with security policies defined using the UCON$_{ABC}$ framework annotated within the business process model. The integration of the business process model, the data model, and the security policies provides a context where more complex reasoning can be applied about the satisfiability of the security policies in accordance with the business process and data models. To do so, we transform the original models, including security policies, into the BAUML framework (an artifact-centric approach to business process modelling). Once this is done, it is possible to ensure that there are no inherent errors in the model (verification) and that it fulfils the business requirements (validation), thus ensuring that the business process and the security policies are compatible and that they are aligned with the business security requirements.

Keywords: Business Process, Security policy, Usage control model, Data artifact, Reasoning.

1. Introduction

Business processes specify the workflow of the activities in an organisation facilitating decision-making support [43] to achieve its objectives. These activities are not carried out in a void, but in many cases, they have to follow certain compliance rules which govern the operation of a company [17]. In this respect, compliance rules are also used for risk management [60] to control threats. Thus, a set of compliance rules may refer to security policies to control security threats. According to the SANS Institute definition, a security policy is a set of security requirements or rules (i.e., access control restrictions) that must be met in order to achieve the business goals. Thereby in this paper, we assume that a set of rules may represent a security policy of an organisation. The necessity of including

security policies in business process models is well-known and has been studied in the literature [33], but security issues are mostly overlooked by default and not tackled in a practical way.

Security policies are not necessarily defined at the same time as the business process; rather, they are usually defined and implemented at later stages of software development. As a result, many times they are specified independently from one another. However, security policies and business process must be aligned [2]. Ensuring this will reduce risky situations and the propagation of errors during process deployment [53]. The combination of both business processes and policy rules is fundamental for the Business Continuity, as described in ISO 22301:2012.

Unfortunately, traditional approaches to process modelling are insufficient when it comes to defining security policies. The inclusion of security controls into process-aware information systems is currently an open challenge [26]. Most of the process-centric approaches try to incorporate access control mechanisms by adapting traditional access control models [29]. However, they fail to incorporate the flexible and complex security policies that modern business information systems demand. These process-centric approaches tend to focus on representing the sequence of activities in the process and disregard or place little importance on the data required. However, security policies may refer to complex data, which cannot be represented through the process-centric approaches. Further, security policies can represent restrictions regarding the number of uses of the resources. One such example is found in the context of a customer's loan request to a credit provider [32]. A security requirement could be that *it is not possible to request a loan when there are previously denied loan requests* or *a staff member is not allowed to review more than ten loan requests in a period of time*. Note that, due to the lack of a data model, it would be impossible to represent this policy.

On the other hand, artifact-centric approaches incorporate data in the definition of the process, and are more appropriate when security policies are involved, since it will be possible to represent them. Not only this but following an artifact-centric approach makes it possible to apply reasoning techniques to the process model and the security policies. Through these techniques, it can be checked that the process model and the security policies are aligned, i.e. there are no contradictions between them (verification), and that they fulfil the business requirements (validation).

As mentioned previously, the importance of specifying security-aware business processes is well-known [5]. Moreover, security policy complexity has been studied in [50], but it was only sketched how they could be modelled in an artifact-centric paradigm. In terms of reasoning, existing artifact-centric approaches do not yet consider verification and validation of security policies. For these reasons, there is a need for a proposal that deals with the verification of the security of business processes. As we have explained, it should be based on the use of an artifact-centric approach to be able to represent complex data structures related to security policies. Therefore, the main challenges to tackle are twofold: 1) the specification security policies based on UCON models into artifact-centric process models, and; 2) the provision of reasoning techniques to verify the security policies using an artifact-centric approach.

Summarising, the main contributions of this paper are:

1. **Definition of an enriched model that includes security policies over data artifacts**. We have defined a model which enables the definition of UCON-based secu-

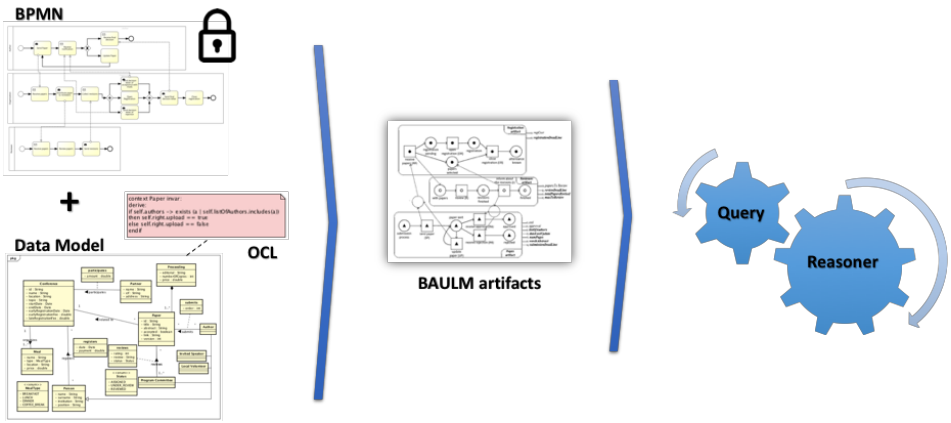


Fig. 1. Proposed Framework

rity policies [39] for artifact-centric process models. To do this, we use BPMN [36] and UML class diagram [37], which are *de facto* standards for process modelling and data specification respectively. We combine and enrich both notations by using security policies defined in Object Constraint Language (OCL) [10] following the UCON$_{ABC}$ model. This provides a data artifact process model with security policies defined in OCL.

2. **Transformation of partial models into the BAUML framework**. We have defined and implemented a transformation of the initial models into a BAUML framework [15] which can be used for reasoning on the model, detecting potential errors and ensuring that it fulfils the requirements and goals.
3. **Reasoning using the enriched model**. After a transformation process, we propose to reuse existing techniques [15] for verifying and validating the artifact-model as a whole, considering also the security policies and the data involved in the policies into a BPMN model.
4. **Evaluation of feasibility.** We have used a running example as a Proof-of-Concept (PoC) during the explanation in each stage and to demonstrate how our approach can reach the verification of a security policy.

The remainder of the paper is structured as follows. Section 2 presents the enriched model and the different parts that form it with an example. Section 3 details how these initial models can be translated into the BAUML framework for reasoning. Section 4 presents the types of reasoning that can be tackled thanks to using the enriched model. Section 5 analyses the related work. Finally, Section 6 presents the conclusions and further work.

2. Enriching Process Models with Security Policies

This section presents the models used in our approach: the UML class diagram, the BPMN diagram, OCL operation contracts and the UCON$_{ABC}$ framework to represent security

policies. In addition, we introduce a running example, to make our proposal easier to understand.

The UML class diagram is used to represent the data in the domain of interest, and the BPMN diagram to model the process. Both models are interrelated in so far as UML diagrams are able to represent the data and their relations while BPMN provides an activity-centric perspective about the activities that can make changes to the data. Both languages are the standard and most common formalisms for representing data and processes, respectively. In addition, we use OCL operation contracts to formally specify each task in the business process, similarly to [12, 38]. This provides the ability for reasoning or executing the resulting models. We then enrich the models including security policies defined using the UCON$_{ABC}$ framework.

To illustrate our approach, a running example based on the customer's loan request [32] is used through the paper. The running example consists of a loan request to a credit provider which considers two acceptance reports before deciding on the request.

2.1. UML Class Diagram

A UML class diagram is formed of a set of classes (or concepts), which may be in a hierarchy, n-ary associations among such classes (where some of them might be reified, i.e, association classes), and some attributes inside these classes. In addition, a UML schema might be annotated with minimum/maximum multiplicity constraints over its association-ends/attributes, and hierarchy constraints (i.e., disjoint/complete constraints).

Figure 2 shows the UML class diagram representing the data required by the process in our running example. For example, a *LoanRequest* is defined by its *id*, *amount*, *pending*, *accepted*, *date* and *risky*. *Pending* and *accepted* attributes represent whether the loan is waiting for approval or has been accepted, respectively. In turn, a *Customer* may submit several loan requests, which are going to be revised by the *Operation Staff* of the *Credit Provider* to decide on the *risk* and the *rate* of the loan.

2.2. BPMN Diagram

BPMN (Business Process Model and Notation) is a widely used and well-known ISO and OMG standard language for modelling business processes known as the defacto standard for business process modelling [25]. In a nutshell, the language uses nodes to represent the activities or tasks of the process, whose execution order is determined by a set of directed edges. Different gateway nodes are available to control the flow, to allow for parallel or alternative execution paths, for instance. Moreover, using BPMN it is also possible to represent the interaction between different parties involved in the process, messages and business objects are able to flow between various business processes [41].

The BPMN model is shown in Figure 3 where three different pools cover its main functions: pool *Customer* manages the request process from the viewpoint of a customer; pool *Credit Provider* describes how the administration staff manages the loan request, obtains the acceptance reports and notifies customers about the decisions; and pool *Operation Staff* deals with the evaluation of the loan requests. We use in the example some collaboration components since various data with different cardinalities flow through business process instances [40, 31]. Single instances of the process *Credit Provider* need to

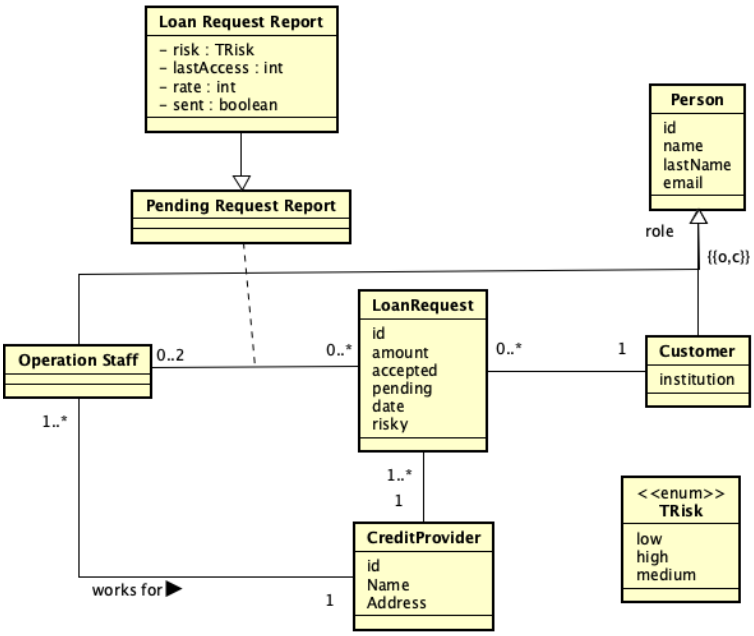


Fig. 2. Data Model of the Loan Request

interact and be synchronised with multiple instances of processes *Customer* and *Operation Staff* simultaneously [18]. Loop Activities (↻) and Parallel Activities (|||) are included to describe the synchronisation between the pools. Data objects, such as *LoanRequest* or *Customer*, appear in the BPMN diagram but their details are modelled in the UML class diagram, as we have shown.

2.3. OCL Operation Contracts

In order to define the behaviour of the tasks that perform work in the BPMN model, we propose the use of OCL operation contracts. Each contract contains: a header, including the operation name and input parameters; a precondition, stating the conditions that *must* be true before the task can be executed; and a postcondition, describing the state of the system *after* the successful execution of the activity. Below we present the contracts of two tasks of the example, *RequestALoan* and *SendApprovedNotification*. OCL contracts for the other activities would be defined similarly.

```
RequestALoan(pId: String, am: int, c: Customer, t: Date, cp: CreditProvider,
     r:boolean)
post: LoanRequest.allInstances()->exists(l | l.oclIsNew() and l.id = pId and
     l.amount = am and l.date = t and l.accepted = false and l.pending = true and
     l.creditProvider = cp and l.customer = c and l.risky=r)
```

Activity *ReceiveNotification* has no OCL operation contract because it waits until receiving a message through the incoming message flow.

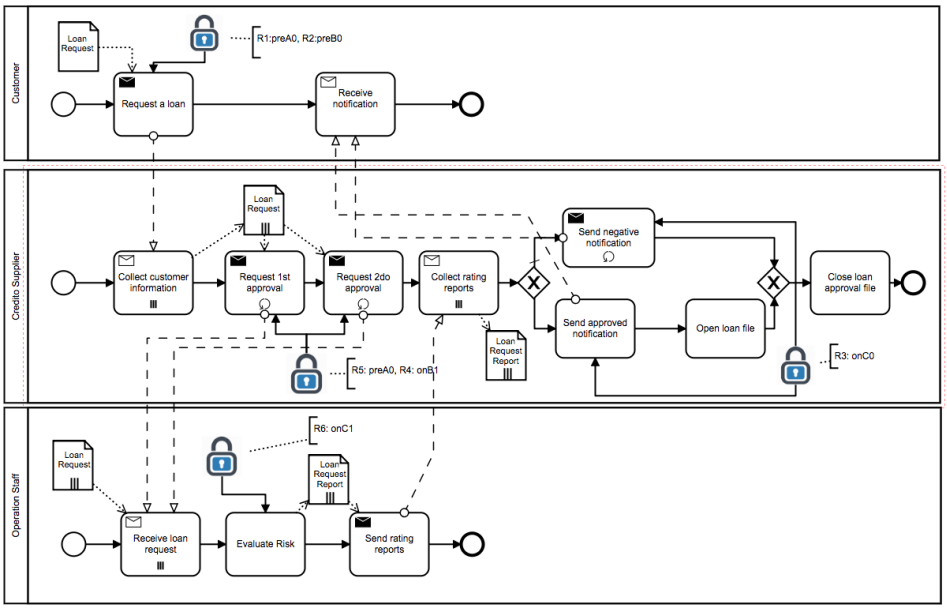


Fig. 3. BPMN Model for the Loan Request

```
SendApprovedNotification(l: LoanRequest, acc: boolean)
post: l.accepted = acc and l.pending = false
```

2.4. Describing Security Policies

Finally, the last elements of our proposal are the security policies. They can be seen as business compliance rules [17] with specific security semantics, such as Separation of Duties (SoD) [9]. There is not a standard framework or formalism to specify them. In this paper, we follow the UCON$_{ABC}$ model [39] which has emerged as a generic formal model to represent complex, adaptable and flexible security policies in new environments, such as Internet of Things (IoT). For instance, Digital Right Management (DRM) is an access control mechanism which can be modelled by UCON$_{ABC}$. Moreover, other traditional access control and trust management mechanisms can be defined by using this model. A UCON$_{ABC}$ model consists of the following components:

- A *Subject* is a component which holds or exercises certain rights on objects. An *Object* is an entity that a subject can access or use with certain rights.
- *Rights* are privileges that a subject can hold and exercise on an object.
- Predicates for the evaluation:
  - *Authorisations* (A) have to be evaluated for usage decisions and return whether the subject (requester) is allowed to perform the requested rights on the object or not.
  - *Obligations* (B) represent functional predicates that verify mandatory requirements a subject has to perform before or during a usage exercise.

- *Conditions* (C) evaluate environmental or systems factors to check whether relevant requirements are satisfied or not.

All these predicates can be evaluated before or while the rights are exercised. For this reason, the UCON$_{ABC}$ model splits each predicate into two types of sub-predicates depending on when it must be evaluated: (1) a pre-Authorisation (*preA*) predicate is evaluated before a requested right is exercised; and (2) an on-Authorisation (*onA*) predicate is checked while the right is exercised. Likewise, obligations and conditions can be divided into pre- and on-predicates. Further, UCON$_{ABC}$ introduces a new factor to be considered in predicates: the use of inmutable or upgradeable attributes. Thus, in certain predicates we may need to check the conditions based on the inmutability or the updating of subject and/or object's attributes. To summarise, all types of predicates and updating predicates supported are given in Table 1 as indicated in [39]. For instance, an onA_3 predicate represents a usage control scenario where the access decision is evaluated during beginning the usage and it also requires a post-update of attributes during the access. Furthermore, thanks to the use of UCON, it enables us to represent traditional access control (e.g., MAC) by means of $preA_0$ and $preA_1$ predicates.

Table 1. The predicates supported by $UCON_{ABC}$ [39].

	0 (inmutable)	**1 (pre-update)**	**2 (ongoing-update)**	**3 (post-update)**
preA	✓	✓	✗	✓
onA	✓	✓	✓	✓
preB	✓	✓	✗	✓
onB	✓	✓	✓	✓
preC	✓	✗	✗	✗
onC	✓	✗	✗	✗

The policies according to [50] are shown in Figure 4. Although our approach can support most of the predicates, just some of them have been used in the running example. They have been modelled together with the BPMN model (cf. Figure 3). The reason behind this is that these policies can help to improve the documentation, the analysis and optimisation of the process model, and the alignment of systems according to the given security requirements, as proposed in [34].

The BPMN model has been designed in an extension of the bpmn.io modeller which integrates a security DSL [52] which enables the graphical specification of security policies employing locks (cf. locks in the diagram) attached to activities. Although the type of rules have been pointed out as text annotations attached to each security policy.

For each rule, the left-hand side of the double implication refers to the *right*, whereas the right-hand side states what needs to be evaluated. For example, policy *onB*$_1$ (cf. R4) states that the staff can only review a maximum of 10 loan requests. Some rules use a *preUpdate* predicate in the right-hand side. It establishes an update of an object's attribute prior to the usage. For instance, *onA*$_1$ (cf. R6) checks if the rate of a loan requests is greater or equal than six and the risk assigned is *low* or *medium*. In terms of subjects and objects, *onC*$_0$ (cf. R4) is an *on condition* (*C*), and its subject is the credit provider and its objects are a loan request and a loan request report.

For a better understanding, a brief description of the security policies is given below:

R1. $\boldsymbol{preA_0}$: $\{sendrequest \leftrightarrow \ \forall l \in self.customer.loanRequest \mid (l.pending = fasle \wedge l.accepted = true))\}$
R2. $\boldsymbol{preB_0}$: $\{sendrequest \leftrightarrow (self.amout \leq 50{,}000\ and\ self.risky = false) \vee (self.amout > 50{,}000\ and\ self.risky = true)\}$
R3. $\boldsymbol{onC_0}$: $\{approveloan \leftrightarrow \forall l \in LoanRequest.PendingRequestReport \mid (l.IsTypeOf(LoanRequestReport)) \wedge (self.risk =' low' \vee self.risk = 'medium')\}$
R4. $\boldsymbol{onB_1}$: $\{reviewrequest \leftrightarrow \ preUpdate(\#self.operationStaff.loanRequest) \wedge \#self.operationStaff.loanRequest \leq 10\}$
R5. $\boldsymbol{preA_0}$: $\{reviewrequest \leftrightarrow \ \exists l \in LoanRequestReports\ (l.loanrequest.id = self.loanrequest.id) \wedge l.operationStaff.id \neq self.operationStaff.id\}$
R6. $\boldsymbol{onA_1}$: $\{sendreport \leftrightarrow \ preUpdate(self.rate) \wedge self.rate \geq 6 \wedge self.loanRequest.amount \leq 5000 \wedge (self.risk =' low' \vee self.risk =' medium')\}$

Fig. 4. Security Policies for the Loan Request

1. **R1** : $preA_0$: the request of more than one loan is not permitted when there are previously unaccepted loans.
2. **R2** : $preB_0$: the loan requests of more than fifty thousand are not permitted for the credit provider without accepting a clause of risk.
3. **R3** : onC_0: the loan request is accepted iff risk is medium or low.
4. **R4** : onB_1: an operation staff cannot review more than ten loan requests at the same time.
5. **R5** : $preA_0$: the second review must not be the same than the first one (Separation of Duties principles).
6. **R6** : onA_1: the reviewer is able to send a loan request report with a rate of six (or greater), an amount less than five thousand, and risk low or medium since requires supervision (Four eye principle).

3. Transforming the Models into an Integrated Solution

Given the models described in the previous section, our goal is to determine the correctness of the business process model as a whole (i.e. considering the data and process models, and the definition of the activities) and its security policies. This means checking that there are no errors, and that the requirements are fulfilled when the models and security policies are considered together, basing our reasoning approach on [15].

Three steps need to be carried out to achieve this objective:

1. Formalize the security policies, so that they can be incorporated into the models.
2. Transform our starting models to be able to reason with them.
3. Perform the reasoning itself, after merging the formalized security policies with the model.

The first two steps are described in the remainder of this section. Step 3 is explained in section 4.

3.1. Formalizing Security Policies Utilizing OCL Constraints

The security policies defined in Figure 4 give an intuitive idea of their meaning, but as they are, cannot be added to the model for reasoning due to a lack of formalization. To solve this, we specify them using OCL language. This is not a limitation since the transformation from informal models to specific formal models has been tackled in previous works [42] and using OCL as a formalism to specify $UCON_{ABC}$ policies has been considered in previous work [27].

Each security policy defines conditions over a set objects. We propose representing a security policy as follows:

```
<SecPolicyName>
Objects: <obj1>:<Type1>, ..., <objN>:<TypeN>
Condition: <OCL expression>
```

where `OCL expression` refers to $\texttt{obj}_1$ to $\texttt{obj}_N$ using OCL constructs and should result in a Boolean value. Note that these objects should either be input parameters of the tasks to which the policies are attached or be created by them.

We also allow the use of `@post` in `OCL expression`, to refer the new value of an attribute. This is necessary for policies that include the expression *preUpdate*, indicating that the new value of the element should be considered. Below, we show policies $R1$, $R2$, $R4$ and $R6$ expressed in OCL:

R1 ($preA_0$):

```
Objects: c:Customer
Condition: c.loanRequest->forAll(l | l.pending=false and l.accepted=true)
```

R2 ($preB_0$):

```
Objects: lr:LoanRequest
Condition: (lr.amount ≤ 50,000 ∧ lr.risky = false) ∨ (lr.amount > 50,000 ∧
     lr.risky = true)
```

R4 (onB_1):

```
Objects: op:OperationStaff
Condition: op.loanRequest@post->size() ≤ 10
```

R6 (onA_1):

```
Objects: l:LoanRequestReport
Condition:  l.rate@post ≥ 6 ∧ l.loanrequest.amount ≤ 5000 ∧ (l.risk = 'low' ∨
     l.risk = 'medium')
```

3.2. Transforming the Models into BAUML

As stated earlier, our goal is to be able to verify UCON_{ABC} policies in the context of a business process model annotated with data artifacts that support complex data structures. To achieve this, we will apply the verification techniques for artifact-centric business process models [15]. In order to do so, we need to adapt our starting models to the input required by the BAUML framework.

BAUML uses four different models: a UML class diagram, a UML state machine diagram, UML activity diagrams and OCL operation contracts. Therefore, we will need to translate or map the starting models (BPMN diagram, UML class diagram, OCL operation contracts and security policies) into these, to be able to reuse the existing techniques. Intuitively, there will be an (almost) direct mapping between the class diagrams and the OCL operation contracts in both approaches. However, we will need to translate the BPMN diagram into a state machine diagram and a set of activity diagrams to obtain an equivalent BAUML model. For this reason, we introduce state machine and activity diagrams.

Definition 1. *A state machine diagram is defined as* $S_A = \langle V, v_o, v_f, E, X, T \rangle$*, where* V *is a set of states,* $v_o \in V$ *is the initial state,* $v_f \in V$ *is the final state,* E *is a set of events,* X *is a set of effects, and* $T \subseteq V \times OCL_{\mathcal{M}} \times E \times X \times V$ *is a set of transitions between pairs of states, where* $OCL_{\mathcal{M}}$ *is an OCL condition over* $\mathcal{M}$ *that must be true in order to the transition to take place. Note that* v_o *cannot have any incoming transition, and* v_f *cannot have any outgoing transition.*

Definition 2. $\mathcal{P}$ *is a set of UML activity diagrams, such that for every state machine diagram* S=$\langle V, v_o, v_f, E, X, T \rangle \in \mathcal{S}$*, and for every event* $\varepsilon \in$ EXTEVENTS(S) *there exists exactly one activity diagram* $P_\varepsilon \in \mathcal{P}$*.* P_ε *is a tuple* $\langle N, n_o, n_f, F \rangle$*, where* N *is a set of nodes,* $n_o \in N$ *is the initial node,* $n_f \subset N$ *is the set of final nodes and* F *is a set of transitions between pairs of nodes.*

Obtaining an equivalent BAUML model The main challenge is to translate the BPMN diagram into a state machine diagram and a set of activity diagrams. This is not a trivial task since the former shows the interaction among the evolution of different classes in the class diagram, whereas in the BAUML modelling approach this interaction is implicitly represented using state machine diagrams. Other approaches also tackled this type of problem as in [14] where the authors propose the synthesising of object life cycles (state machines) from business process models.

Due to this complexity we deal here with a fragment of the BPMN diagram. In particular, we will translate only one of the pools, the *Customer* one. We focus on this pool because its tasks have a direct effect on the evolution of class *LoanRequest*, as shown on the contracts of activities or tasks *RequestALoan* and *ReceiveNotification*.

For this purpose, we will distinguish two types of tasks in the BPMN diagram:

- Tasks that send information or perform certain work by themselves. They can be identified by the dark envelope or by the lack of a symbol. We will refer to them as *action tasks*.
- Tasks that receive information and, as such, they are waiting for something to happen outside the scope of the pool. They can be identified by a white envelope symbol. We will call them *receive message tasks*. If these tasks do not have an incoming message flow, we will refer to them as *passive tasks*.

The first type of task will correspond to events in the state machine diagram, whereas the second type to states, but will require the incoming message flows to be considered in the translation process. Moreover, XOR-split nodes will also correspond to states. We will globally refer to XOR-split nodes, initial nodes and passive tasks as *passive nodes*.

Obtaining the State Machine Diagram. Algorithm 1 begins the translation process by obtaining a list of all the nodes in the BPMN diagram and translating them into the corresponding element in the state machine diagram. Note that the algorithm merely translates the nodes and not the connections between them.

Action tasks will correspond to events E. Passive tasks, XOR-split nodes and the final node correspond to states V. Initial nodes correspond to the initial pseudo-state v_o of the state machine diagram. XOR-merge nodes do not correspond to a specific node in the resulting state machine diagram. Finally, each incoming message flow will correspond to an event.

Algorithm 1 translateNodesAndMessages()

```
nodeMap = ∅
▷ We first create a map containing the task nodes in the BPMN diagram and their translation to an event or a state.
▷ nodeList contains all the nodes in the BPMN diagram
for all node ∈ nodeList do
    if node is ActionTask then
        nodeMap.add(<node, new Event(node)>)
    else if node is PassiveTask then
        nodeMap.add(<node, new State(node)>)
    else if node is XOR-split then
        nodeMap.add(<node, new State(node)>)
    else if node is InitialNode then
        nodeMap.add(<node, new InitialPseudostate(node)>)
    else if node is FinalNode then
        nodeMap.add(<node, new State(node)>)
    end if
end for
▷ We then create an event for each incoming message flow in the pool
▷ receiveMessageTaskList contains all the tasks with incoming message flows
▷ messageMap will contain a map between the receive message task and the incoming message flows, translated to
events
messageMap = ∅
for all rm ∈ receiveMessageTaskList do
    incomingFlowList = rm.getIncomingMessageFlows()
    eventList = ∅
    for all if ∈ incomingFlowList do
        sourceNode = if.getSource()
        event = new Event (sourceNode)
        eventList.add(event)
    end for
    messageMap.add(<rm, eventList>)
end for
```

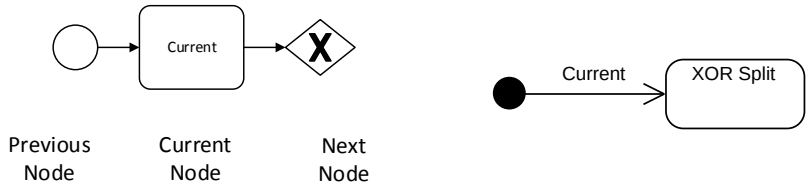
Algorithm 2 initiates the processing of the nodes. It iterates over all the nodes and obtains the next nodes for the current node. Then it provides the current node, the next nodes and the node map (obtained by Algorithm 1) to Algorithm 3.

Algorithm 2 translateToSMD()

```
▷ nodeList contains all the nodes in the BPMN diagram
▷ nodeMap corresponds to the nodeMap obtained previously
for all node ∈ nodeList do
    ▷ We go through the elements of the list in order (i.e. before the processing of a node all its previous nodes must have
been processed)
    nextNodeList = node.getNextNodes()
    ▷ getNextNodes() ignores XOR-merge nodes and returns the targets of the XOR-merge
    if !nextNodeList.isEmpty() then
        for all nextNode ∈ nextNodeList do
            processNode(node, nextNode, nodeMap)
        end for
    end if
end for
```

Algorithm 3 is executed for every node (and its next node) in the pool of interest in the initial BPMN diagram. As input, the algorithm receives the following: the current node ($node$), the next node ($nextNode$), and the node map ($nodeMap$), which contains the correspondence between the nodes in the BPMN diagram and the state machine diagram, previously created by Algorithm 1. The algorithm assumes that all the nodes that

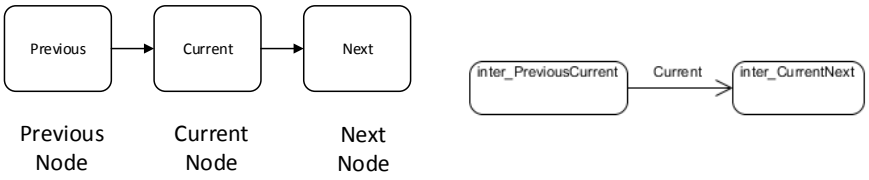
can be executed previous to the current node have already been translated and connected properly.



(a) BPMN fragment (b) Translation of fragment in (a)

Fig. 5. Translation of an action node surrounded by passive nodes

Then, it translates the connections between the current and the previous/next nodes according to their types. If the current node is an action task, it will correspond to an event in the state machine diagram. Hence, we will have to create a transition with the event, which will require a source and a target state. These source and target states will correspond to other BPMN nodes, if the surrounding nodes are passive nodes or a message receive task (see Figure 5). In contrast, if the surrounding nodes are action tasks, they will require an auxiliary state (see Figure 6).

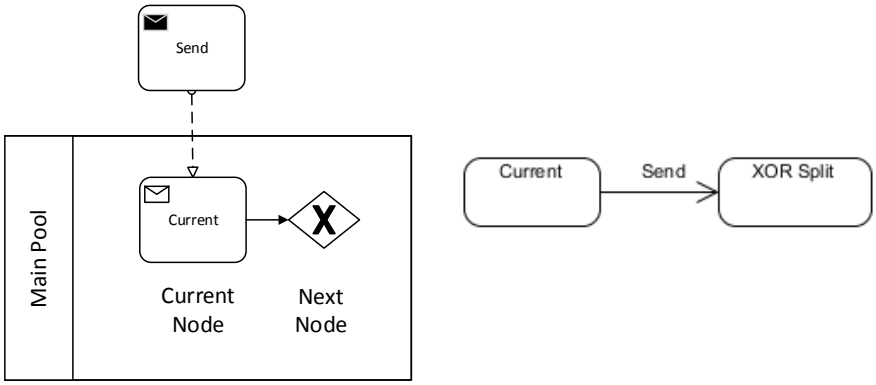


(a) BPMN fragment (b) Translation of fragment in (a)

Fig. 6. Translation of an action node surrounded by other action nodes

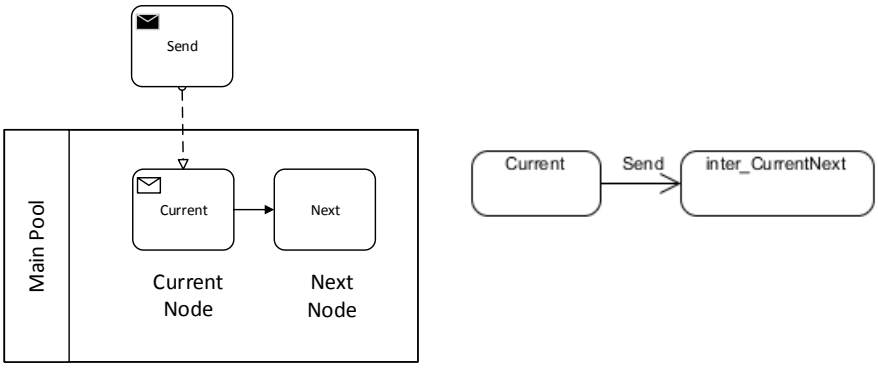
If the current node is a receive message task, the semantics of BPMN state that it is not possible to move to the next node until a message is received. Therefore, for each incoming message flow in the node, the state machine diagram will require a transition with the message represented as an event to move to the next state. Considering this, the current node corresponds to a state acting as the source state of the transition. Then we need to consider the next node. If the next node is another receive message task or a passive node (e.g. XOR-split), then the target state will be the corresponding state in the state machine diagram (see Figure 7). Otherwise, if the next node is an action task , an intermediate state will need to be created to act as the target state (Figure 8).

If the current and next node are passive nodes, the only thing that needs to be done is to create a transition between both (see Figure 9). This transition will be automatic as there will be no events between both. Note that we do not consider the case of the next



(a) BPMN fragment (b) Translation of fragment in (a)

Fig. 7. Translation of a receive message task followed by a passive node



(a) BPMN fragment (b) Translation of fragment in (a)

Fig. 8. Translation of a receive message task followed by an action

node being an action or a receive message task, as in this case the necessary changes will be made by the algorithm in the next iteration.

(a) BPMN fragment (b) Translation of fragment in (a)

Fig. 9. Translation of a passive node followed by another passive node

Algorithm 3 processNode(node, nextNode, nodeMap)

```
if node is ActionTask then
    Event e = nodeMap.get(node)
    prevNodeList = node.getPrevious()
    for all prevNode ∈ prevNodeList do
        if prevNode is PassiveNode then
            State source = nodeMap.get(prevNode)
            if nextNode is PassiveNode ∨ nextNode is ReceiveMessageNode then
                State target = nodeMap.get(nextNode)
                Transition t = new Transition (source, target, e)
            else if nextNode is ActionTask then
                State target = new State ("inter_" + node.getName() + nextNode.getName())
                Transition t = new Transition(source, target, e)
            end if
        else if prevNode is ActionTask then
            if nextNode is PassiveNode ∨ nextNode is ReceiveMessageNode then
                ▷ We obtain the target state of the previous ActionTask, which will be the source state for our new transition
                State source
                State target = nodeMap.get(nextNode)
                Transition t = new Transition (source, target, e)
            else if nextNode is ActionTask then
                ▷ We obtain the target state of the previous ActionTask, which will be the source state for our new transition
                State source
                State target = new State ("inter_" + node.getName() + nextNode.getName())
                Transition t = new Transition (source,target, e)
            end if
        else if prevNode is ReceiveMessageNode then
            State source = get intermediate state generated by Rec. Message Event
            if nextNode is PassiveNode ∨ nextNode is ReceiveMessageTask then
                State target = nodeMap.get(nextNode)
                Transition t = new Transition (source, target, e)
            else if nextNode is ActionTask then
                State target = new State ("inter_" + node.getName() + nextNode.getName())
                Transition t = new Transition (source, target, e)
            end if
        end if
    end for
else if node is ReceiveMessageTask then
    List<Event> eventList = messageMap.getIncomingMessageFlow(node)
    State source = nodeMap.get(node)
    if nextNode is ActionTask then
        State target = new State ("inter_" + node.getName() + nextNode.getName())
        for all e ∈ eventList do
            Transition t = new Transition (source, target, e)
        end for
    else if nextNode is PassiveNode ∨ nextNode is ReceiveMessageTask then
        State target = nodeMap.get(nextNode)
        for all e ∈ eventList do
            Transition t = new Transition (source, target, e)
        end for
    end if
else if node is PassiveNode ∧ nextNode is PassiveNode then
    State source = nodeMap.get(node)
    State target = nodeMap.get(nextNode)
    Transition t = new Transition (source, target)
    ▷ If nextNode is ActionTask, we do nothing because it will be processed in the next iteration.
end if
```

Figure 10 shows the resulting translation for the customer pool in the BPMN diagram. Note that the action task *RequestALoan* corresponds to an event. The receive message task *ReceiveNotification* corresponds to a state. The incoming message flow of *ReceiveNotification* connected action tasks *SendApprovedNotification* and *SendNegativeNotification* from another pool to *ReceiveNotification*. Therefore, these action tasks, *SendApprovedNotification* and *SendNegativeNotification*, has been translated to an event in the state machine diagram. Finally, state *FinalState* corresponds to the final state in the BPMN diagram. We have given it this name for easier readability.

The BAUML framework requires the state machine diagram to be linked to a class, called the *artifact*. In this case, it will be linked to class *LoanRequest*, since the changes made by the activities or tasks have an impact on it.

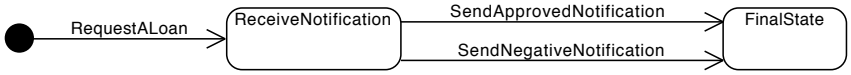


Fig. 10. State machine diagram

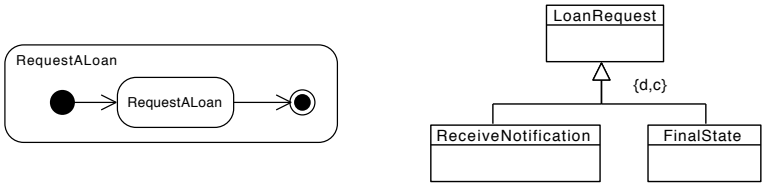


Fig. 11. Activity diagram and updated class diagram

Obtaining the Activity Diagrams The BAUML framework requires an activity diagram for each event in the state machine diagram. They can be automatically generated for each event obtained by the previous algorithms. Therefore, for each event we will have the corresponding activity diagram: it will have an initial node, a final node, and one task with the corresponding OCL operation contract. Figure 11 shows the activity diagram for event *RequestALoan*.

Once the state machine and activity diagrams from the BPMN diagram have been obtained, we already have all the components necessary to reason using the techniques in [15]. What needs to be done is to make some minor changes to the class diagram and the operation contracts before incorporating the security policies into the model.

Updating the Class Diagram and the Operation Contracts The last step in the process is to update the class diagram and the OCL operation contracts. In the BAUML framework, the state machine diagram shows the evolution of an artifact (i.e. a class in the class diagram), and each state corresponds to a subclass of the artifact. Therefore, once the

state machine diagram and the activity diagrams have been obtained, the class diagram needs to be updated by including as many classes as states there are in the state machine diagram. In our particular example, class *LoanRequest* should be updated with subclasses *ReceiveNotification* and *FinalState*, forming a *disjoint*, *complete* hierarchy (see Figure 11).

Then, it is also necessary to update the OCL operation contracts to ensure the proper evolution of the class. Therefore, the postcondition of the contract of task *RequestALoan* must be updated to ensure that the *LoanRequest* has the type *ReceiveNotification*:

```
LoanRequest.allInstances()->exists(l | l.oclIsNew() and l.id = pId and l.amount =
     am and l.date = t and l.accepted = false and l.pending = true and
     l.creditProvider = cp and l.customer = c and l.risky=r and
     l.oclIsTypeOf(ReceiveNotification))
```

Similarly, *SendApprovedNotification* and *SendNegativeNotification* should be updated to change the state of *LoanRequest* to *FinalState* and to ensure it is no longer of the previous subtype (*ReceiveNotification*). Below we show the updated postcondition for *SendApprovedNotification*:

```
l.accepted = acc and l.pending = false and l.oclIsTypeOf(Finalstate) and not
     l.oclIsTypeOf(ReceiveNotification)
```

These updates can be performed automatically, as the changes can be inferred from the state machine diagram obtained previously. More specifically, considering that each task is part of a transition in the state machine diagram, the postcondition should ensure that the class has the subtype represented by a target state, and that it no longer is of the subtype of the source state.

4. Reasoning on Security Policies

Once the security policies have been defined by using OCL and the starting models translated to BAUML, we need to merge both with the goal of checking them as a whole. First, it is necessary to present how to add the policies to the models and later on to explain how reasoning with them.

4.1. Adding the Security Policies to BAUML

Intuitively, if the conditions established by a security policy are not met, the task to which they are linked should never take place. Considering that our modelling approach uses pre and postconditions for each task, this means that security policies should be added to the preconditions, to ensure that the task does not execute if they are not met, with the following considerations:

1. The tasks that these policies are attached to, should have as input the objects of the policy.
2. If this is not the case, these objects should be created or specialised by the task. This is easy to identify by looking for either `obj.oclIsNew()` (creation) or `obj.oclIsTypeOf(ObjType)` (specialization) in the postcondition.

3. If the OCL expression corresponding to the security policy contains `@post`, it needs to be modified before it can be added to the precondition. `@post` refers to the value of the attribute in postcondition time, which cannot be accessed in precondition time. To deal with this, we need to look for the parameter that will assign a new value to this attribute, and substitute it in the expression.

Note that if the task is a *receive message task* in the BPMN, the predicates will be added to the tasks that result from the translation of the incoming message flows to events.

Returning to our example, there are two security predicates that should be checked in the precondition of *RequestALoan*, as indicated by the BPMN diagram in Figure 3. We will first look at `R1`, whose OCL corresponds to the security policy as defined earlier: `c.loanRequest->forAll(l | l.pending=false and l.accepted=true)`. In this case, this expression can be incorporated directly into the precondition of the task. Since the object of the policy is *Customer c* and *RequestALoan* has a customer as input parameter (also `c` in our example), we only need to make sure that the policy refers to this customer, by rewriting its name if necessary.

In the case of `R2`, its object refers to *LoanRequest*. The operation contract does not have a *LoanRequest* as input; however, it does create the *LoanRequest* in its postcondition (`LoanRequest.allInstances()->exists(l | l.oclIsNew()...)`). In this case, we have to replace all references to `lr.amount` and `lr.risky` in the policy with the corresponding parameters which will be accessible at precondition time. Since `l.amount = am` and `l.risky = r`, as stated in the postcondition, we have to replace `lr.amount` with `am` and `lr.risky` with `r`, resulting in `(am<=50000 and r=false) or (am>50000 and r=true)`, as shown below.

```
RequestALoan(pId: String, am: int, c: Customer, t: Date, cp: CreditProvider,
     r:boolean)
pre: c.loanRequest->forAll(l | l.pending=false and l.accepted=true) and ((am<=50000
     and r=false) or (am>50000 and r=true))
post: LoanRequest.allInstances()->exists(l | l.oclIsNew() and l.id = pId and
     l.amount = am and l.date = t and l.accepted = false and l.pending = true and
     l.creditProvider = cp and l.customer = c and l.risky = r)
```

4.2. Reasoning over Security Policies

Once the security policies have been added to the BAUML model it is possible to run verification and validation tests as described in [15]. Given a BAUML model, this approach automatically translates it into the required logic for satisfiability checking, and then determines whether the model fulfils certain semantic correctness properties. The underlying satisfiability checker can deal with negated predicates and, in the case of unsatisfiability, provides the list of constraints that prevent it.

Internally, this is done by SVTe [16], a tool that uses the CQC_E method [47], and which has also been used successfully in [44, 46]. It is aimed at building a consistent state of a database schema that satisfies a certain goal. Starting from an empty solution (i.e. an empty database), and given a goal, the database schema, a set of constraints and derivation rules, it tries to obtain a set of base facts that satisfy the goal without violating the constraints. Note that for BAUML models, the derivation rules include the translation of the tasks, activity diagrams and state machine diagram. The CQC_E method is a semidecision procedure for finite satisfiability. That is, if the solution contains infinite elements,

it does not terminate. However, termination is ensured if the model fulfils a set of properties, explained in [11]. They can be summarized as follows: 1) the cardinalities of the associations, 2) the number of classes should be bounded, 3) the OCL expressions should be unidirectional and navigational, i.e. when dealing with class instances that are modified by the model, they only refer to elements connected to the starting element. During the inference process, CQC_E uses Variable Instantiation Patterns (VIPs), which generate only the facts that are needed to achieve the goal. If there is no instance found, then VIPs guarantee that the goal cannot be achieved. Considering this, SVTe provides two different types of result, depending on the outcome of the reasoning process. On the one hand, if it finds a solution that fulfils the goal, it provides a sample instantiation. On the other hand, if there is no solution, it shows the constraints that prevent the goal's achievement.

As we mentioned earlier, it is possible to carry out both validation and verification tests. Verification tests looks for inherent errors in the model, answering the question "Is the model right?", and validation tests ensure that the model represents the domain appropriately (i.e. it fulfils the requirements), answering the question "Is it the right model?".

Given the BAUML model, verification tests can be generated and performed automatically, as shown by the prototype tool in [15]. Some examples of these are: ensuring the liveliness or a class or an association (i.e. ensuring that instances can be created), looking for redundancies in the integrity constraints or ensuring that tasks in the process model can execute. On the other hand, validation tests require manual definition, although they can be run automatically. For instance: *can loan requests for 60,000 be made without accepting the risky clause?* or *can a customer make a new loan request when some of his other loan requests have been denied?*.

$$
\begin{aligned}
loanReq() &\leftarrow LoanRequest(oid, id, am, ac, pend, d, t, r) \wedge am = 60,000 \wedge r = false \\
newLoanReqWhenDenied() &\leftarrow LoanRequest(oid, id, am, ac, pend, d, t1) \\
&\quad \wedge LoanRequest(oid2, id2, am2, ac2, pend2, d2, t2) \wedge pend2 = false \\
&\quad \wedge ac2 = false \wedge oid \neq oid2 \wedge t1 = t2
\end{aligned}
$$

These tests have the form of logic derivation rules. Each test has a head: if it is possible to generate the head of the rule, then the test is satisfiable. The body contains the representation of the elements in the model and the conditions which they should satisfy for the test.

Returning to the validation tests above, *LoanRequest* corresponds to an instance of class *LoanRequest*, and $am = 60,000 \wedge r = false$ state the value that these variables am, r should have. The second example states that, in order for the test to execute successfully, there must be two different *LoanRequests* which coexist simultaneously ($t1 = t2$), one of which has been denied.

If we run these tests, the first one will result in unsatisfiability, whereas the second test will execute successfully. In the first case, this is due to security policy `R2`. In the second case, although there is policy `R1` to prevent the creation of new *LoanRequests* when previous ones have been denied, the policy does not consider the case in which several pending loan requests from the customer may coexist simultaneously. Eventually, one of these loan requests may be denied, but there will be other pending loan requests which will require evaluation and which may be accepted.

5. Related Work

Compliance of business processes with security policies at the design and runtime stages has been considered in several stages of business process management [26]. The extension of BPMN with annotations related to security requirements is not new [45]. A vast number of works provide several ways to represent and verify security requirements at the design stage, such as [48, 57, 49]. Salnitri et al. [49] establishes mechanisms to represent security policies in BPMN by means of the extension SecBPMN. The authors also provide a language to verify if the business process model complies with the security policies established. [53] proposes a risk analysis of the business process models combining the partial risks of the activities that conform them. However, our approach focuses on the verification of the security policy in artifact-centric business process models to support the contexts where there various involved data and with different cardinalities between them involved in the policy rules. [3] detected the difficulty to combine data and business processes, but different data objects with various relations between them were not included.

Access control models and process-centric approaches. There exist several and diverse access control models, a good taxonomy is provided in [29]. The most prominent traditional access control models encompassed Mandatory Access Control (MAC), Discretionary Access Control (DAC), Role-based Access Control (RBAC), and; Attribute-based Access Control (ABAC). These access control models are based on the restriction on performing certain right. Depending on the properties used to evaluate the conditions to apply or not the restriction define the access control model. For instance, MAC and DAC are based on the ownership property but RBAC is based on the role property. There exist also some access control models that are emerged as extension for workflow systems such as TBAC [58][57], T-RBAC [35], and W-RBAC [55]. These access control models just consist of adapting RBAC to the process-centric systems. However, all these access control models focus on the request and grant the access only at request time skipping other usage control aspects such as the number of times of using resources/objects or the time of using. In conclusion, these access control models fail to integrate sophisticated decision-making as UCON does.

Currently, **monitoring and process mining techniques** are new trends in order to detect whether certain security requirements are fulfilled by analyzing event logs [1, 6]. The work in [51] enables the generation of security configuration workflows, whereas [54] provides a framework to design product lines to verify security policies in accordance to a set of available configurations. Nevertheless, these works only consider the activity-centric perspective and overlook the artifact-centric perspective in most cases. In terms of complying with security policies on activity-centric process models, different formalisms are proposed to define security compliance rules, such a logic approach based on LTL or declarative approach based on Answer Set [23]. The best approach for large-scale security policies verification depends on time-consuming and easy to understand/update. Not only security policies are important, the context-awareness of the business process should also be considered (e.g. one of the process's instances is executed in a platform and another instance is executed in a different platform). For this reason, a recent work proposes a language for the specification and design of context-aware and secure workflow systems [59]. The approaches to enforce security policies at runtime mainly focus on integrating security control mechanisms into business process management systems [9].

Compliance and security policy verification in artifact-centric business processes has been addressed in a previous work [28]. The authors extend the artifact-centric framework by including the modelling of compliance rules, and obtain a model that complies by design. Also, the paper [31] checks for conformance between process models and data objects at design time. Weak conformance is used to verify that the correct execution of a process model corresponds to a correct evolution of states of the data objects. Reachability and weak-termination are verified in artifact-centric models combining structural and data information [8].

Some previous works indicate the difficulty of security compliance using artifact-centric models since there exists no well-defined operational semantics for directly executing the defined models. But the work of [48] has been proposed to support process-aware secure systems modelling and automated generation of secure artifact-centric implementations.

In [50], an extension of artifact-centric process models based on the Usage Control Model introduces mechanisms to specify security policies and verify their correctness. However, it focuses on reachability and weak-termination of the model as a whole.

Verification and validation in artifact-centric business process models is also a related area of research. There are several works which focus on reasoning on Data-Centric Dynamic Systems (DCDSs) [21, 22], grounded on logic. However, DCDSs use condition-action rules and actions defined in logic. In contrast to our approach, these models are not as intuitive, and using condition-action rules changes the representation paradigm, in the sense that they do not force the execution of actions in a certain order. [30] encodes actions in the same way as DCDSs, but it uses a relational database for the data and Petri nets to establish the execution order. In terms of reasoning, the approach mainly targets reachability and model checking of properties defined in first-order logic. Decidability is achieved by state-boundedness.

Other approaches, such as [7, 4, 13], define the models and the properties to be checked in languages derived from logic. As a result, the models under consideration are formal, but they are not intuitive nor practical from the point of view of the business. Similarly, the properties to be checked have to be defined manually.

A more business-friendly representation for artifact-centric business processes can be found on the Guard-Stage-Milestone (GSM) approach. GSM models show the stages in the evolution of an artifact and the guard conditions which have to be true to enter a certain state. However, they also have the concept of milestone: a condition that, once it is achieved, it closes a state [24, 19, 20]. [24] simulates the behaviour of the system given certain data. [19] is able to reason on the models but data types are limited and it only allows one instance per artifact. Finally, [20] performs model checking from a multi-agent perspective, but the number of objects is bounded which may lead to unreliable results when the bound is exceeded.

The approach in [56] uses BPMN diagrams whose tasks may be annotated with pre-conditions and effects defined in logic, and use an optional ontology to define the underlying data. They have a prototype tool that can perform some tests. Since both the ontology and the details of the effects are not compulsory, the final results can only be partial.

Note also that none of these works take security policies into consideration as we do here.

6. Conclusions and Further Work

We have proposed in this paper a combined business process model that supports the definition, verification and reasoning of security policies involving different kinds of data objects. The enriched model consists of a BPMN model, a UML class diagram, OCL operation contracts for the BPMN activities, and UCON$_{ABC}$ security policies defined in OCL. All these components are then automatically translated into a BAUML model which supports a set of verification and reasoning techniques. Thanks to our proposal, organisations are able to describe their security policies into artifact-centric approaches for business process modelling, providing a mechanism for verifying and validate the model's correctness. The model provides an easier manner to include the security rules into business processes and allows us to ensure that they are compatible with the business requirements and goals.

Although our proposal meets the objectives stated in the introduction, also presents some limitations: 1) regarding UCON support, our approach demonstrates the use of only certain types of predicates and update predicates; 2) regarding the reasoning, it is only limited to the satisfactibility or not of a policy from a pool-by-pool perspective, we can go an step forward by considering several pools, and 3) regarding the tools, our approach depends on the use of different tools separately, and it would be interesting to integrate them. Assuming these limitations, as future work we plan to extend the proposal by including new use cases to fully test the whole set of security policies supported by UCON, and to improve the reasoning by considering various pools simultaneously.

Acknowledgments. This work has been supported by Project PID2020-112540RB-C44 funded by MCIN/AEI/ 10.13039/501100011033, Project TIN2017-87610-R funded by MCIN/AEI/10.13039/501100011033 and FEDER "Una manera de hacer Europa", Project 2017-SGR-1749 by the Generalitat de Catalunya, Projects COPERNICA (P20_01224) and METAMORFOSIS by the Junta de Andalucía.

References

1. Accorsi, R., Wonnemann, C., Stocker, T.: Towards Forensic Data Flow Analysis of Business Process Logs. In: 2011 Sixth International Conference on IT Security Incident Management and IT Forensics. IEEE (may 2011)
2. Ahmed, N., Matulevicius, R.: Securing business processes using security risk-oriented patterns. Computer Standards & Interfaces 36(4), 723–733 (2014), https://doi.org/10.1016/j.csi.2013.12.007
3. Alizadeh, M., Lu, X., Fahland, D., Zannone, N., van der Aalst, W.M.P.: Linking data and process perspectives for conformance analysis. Computers & Security 73, 172–193 (2018)
4. Belardinelli, F., Lomuscio, A., Patrizi, F.: Verification of deployed artifact systems via data abstraction. In: Kappel, G., Maamar, Z., Nezhad, H.R.M. (eds.) ICSOC 2011. LNCS, vol. 7084, pp. 142–156. Springer (2011)
5. Bentounsi, M., Benbernou, S., Atallah, M.J.: Security-aware business process as a service by hiding provenance. Computer Standards & Interfaces 44, 220–233 (2016), https://doi.org/10.1016/j.csi.2015.08.011
6. Bezerra, F., Wainer, J., van der Aalst, W.M.P.: Anomaly detection using process mining. In: Halpin, T., Krogstie, J., Nurcan, S., Proper, E., Schmidt, R., Soffer, P., Ukor, R. (eds.) Enterprise, Business-Process and Information Systems Modeling. pp. 149–161. Springer Berlin Heidelberg, Berlin, Heidelberg (2009)

7. Bhattacharya, K., Gerede, C.E., Hull, R., Liu, R., Su, J.: Towards formal analysis of artifact-centric business process models. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) BPM 2007. LNCS, vol. 4714, pp. 288–304. Springer (2007)
8. Borrego, D., Gasca, R.M., Gómez-López, M.T.: Automating correctness verification of artifact-centric business process models. Information & Software Technology 62, 187–197 (2015)
9. Brucker, A.D., Hang, I., Lückemeyer, G., Ruparel, R.: SecureBPMN: modeling and enforcing access control requirements in business processes. In: Atluri, V., Vaidya, J., Kern, A., Kantarcioglu, M. (eds.) 17th ACM Symposium on Access Control Models and Technologies, SACMAT '12, Newark, NJ, USA - June 20 - 22, 2012. pp. 123–126. ACM (2012), https://doi.org/10.1145/2295136.2295160
10. Cabot, J., Gogolla, M.: Object constraint language (ocl): A definitive guide. In: Bernardo, M., Cortellessa, V., Pierantonio, A. (eds.) Formal Methods for Model-Driven Engineering: 12th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2012, Bertinoro, Italy, June 18-23, 2012. Advanced Lectures. pp. 58–90. Springer Berlin Heidelberg, Berlin, Heidelberg (2012)
11. Calvanese, D., Montali, M., Estañol, M., Teniente, E.: Verifiable UML artifact-centric business process models. In: Li, J., Wang, X.S., Garofalakis, M.N., Soboroff, I., Suel, T., Wang, M. (eds.) CIKM 2014. pp. 1289–1298. ACM (2014)
12. De Giacomo, G., Oriol, X., Estañol, M., Teniente, E.: Linking data and BPMN processes to achieve executable models. In: Dubois, E., Pohl, K. (eds.) CAiSE 2017. LNCS, vol. 10253, pp. 612–628. Springer (2017)
13. Deutsch, A., Hull, R., Li, Y., Vianu, V.: Automatic verification of database-centric systems. ACM SIGLOG News 5(2), 37–56 (2018)
14. Eshuis, R., Van Gorp, P.: Synthesizing object life cycles from business process models. Software & Systems Modeling 15(1), 281–302 (Feb 2016)
15. Estañol, M., Sancho, M., Teniente, E.: Ensuring the semantic correctness of a BAUML artifact-centric BPM. Information & Software Technology 93, 147–162 (2018)
16. Farré, C., Rull, G., Teniente, E., Urpí, T.: Svte: a tool to validate database schemas giving explanations. In: Giakoumakis, L., Kossmann, D. (eds.) DBTest 2008. p. 9. ACM (2008)
17. Gómez-López, M.T., Pérez-Álvarez, J.M., Gasca, R.M.: Compliance validation and diagnosis of business data constraints in business processes at runtime. Information Systems 48, 26 – 43 (2015), http://www.sciencedirect.com/science/article/pii/S0306437914001306
18. Gómez-López, M.T., Pérez-Álvarez, J.M., Varela-Vaca, Á.J., Gasca, R.M.: Guiding the creation of choreographed processes with multiple instances based on data models. In: BPM 2016 International Workshops, Revised Papers. pp. 239–251 (2016)
19. Gonzalez, P., Griesmayer, A., Lomuscio, A.: Verifying gsm-based business artifacts. In: Goble, C.A., Chen, P.P., Zhang, J. (eds.) 2012 IEEE 19th International Conference on Web Services, Honolulu, HI, USA, June 24-29, 2012. pp. 25–32. IEEE Computer Society (2012)
20. Gonzalez, P., Griesmayer, A., Lomuscio, A.: Model checking gsm-based multi-agent systems. In: Lomuscio, A., Nepal, S., Patrizi, F., Benatallah, B., Brandic, I. (eds.) ICSOC 2013 Workshops. LNCS, vol. 8377, pp. 54–68. Springer (2013)
21. Hariri, B.B., Calvanese, D., Giacomo, G.D., Deutsch, A., Montali, M.: Verification of relational data-centric dynamic systems with external services. In: Hull, R., Fan, W. (eds.) PODS 2013. pp. 163–174. ACM (2013)
22. Hariri, B.B., Calvanese, D., Giacomo, G.D., Masellis, R.D., Felli, P., Montali, M.: Verification of description logic knowledge and action bases. In: Raedt, L.D., Bessiere, C., Dubois, D., Doherty, P., Frasconi, P., Heintz, F., Lucas, P.J.F. (eds.) ECAI 2012. Frontiers in Artificial Intelligence and Applications, vol. 242, pp. 103–108. IOS Press (2012)
23. Hewett, R., Kijsanayothin, P., Bak, S., Galbrei, M.: Cybersecurity policy verification with declarative programming. Applied Intelligence 45, 83 – 95 (2016)

24. III, F.F.T.H., Boaz, D., Gupta, M., Vaculín, R., Sun, Y., Hull, R., Limonad, L.: Barcelona: A design and runtime environment for declarative artifact-centric BPM. In: Basu, S., Pautasso, C., Zhang, L., Fu, X. (eds.) ICSOC 2013. LNCS, vol. 8274, pp. 705–709. Springer (2013)
25. Kocbek, M., Jost, G., Hericko, M., Polancic, G.: Business process model and notation: The current state of affairs. Comput. Sci. Inf. Syst. 12(2), 509–539 (2015), https://doi.org/10.2298/CSIS140610006K
26. Leitner, M., Rinderle-Ma, S.: A systematic review on security in Process-Aware Information Systems – Constitution challenges, and future directions. Information and Software Technology 56(3), 273–293 (mar 2014)
27. Li, M., Wang, H.: Specifying usage control model with object constraint language. In: 2010 Fourth International Conference on Network and System Security. pp. 391–397 (Sept 2010)
28. Lohman, N.: Compliance by design for artifact-centric business processes. In: BPM 2011 LNCS vol 6896 Springer. p. 99–115 (2011)
29. Majumder, A., Namasudra, S., Nath, S.: Taxonomy and classification of access control models for cloud environments, chap. 2, pp. 23–53. Springer London, London (2014)
30. Masellis, R.D., Francescomarino, C.D., Ghidini, C., Montali, M., Tessaris, S.: Add data into business process verification: Bridging the gap between theory and practice. In: Singh, S.P., Markovitch, S. (eds.) Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA. pp. 1091–1099. AAAI Press (2017), http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14627
31. Meyer, A., Pufahl, L., Batoulis, K., Fahland, D., Weske, M.: Automating data exchange in process choreographies. Inf. Syst. 53, 296–329 (2015)
32. Mpardis, G., Kotsilieris, T.: Bank loan processes modelling using bpmn. In: 2010 Developments in E-systems Engineering. pp. 239–242 (Sept 2010)
33. Müller, G., Accorsi, R.: Why are business processes not secure? In: Fischlin, M., Katzenbeisser, S. (eds.) Number Theory and Cryptography - Papers in Honor of Johannes Buchmann on the Occasion of His 60th Birthday. Lecture Notes in Computer Science, vol. 8260, pp. 240–254. Springer (2013), https://doi.org/10.1007/978-3-642-42001-6_17
34. Neubauer, T., Klemen, M.D., Biffl, S.: Secure business process management: A roadmap. In: Proceedings of the The First International Conference on Availability, Reliability and Security, ARES 2006, The International Dependability Conference - Bridging Theory and Practice, April 20-22 2006, Vienna University of Technology, Austria. pp. 457–464. IEEE Computer Society (2006), https://doi.org/10.1109/ARES.2006.121
35. Oh, S., Park, S.: Task-role based access control (T-RBAC): an improved access control model for enterprise environment. In: Ibrahim, M.T., Küng, J., Revell, N. (eds.) Database and Expert Systems Applications, 11th International Conference, DEXA 2000, London, UK, September 4-8, 2000, Proceedings. Lecture Notes in Computer Science, vol. 1873, pp. 264–273. Springer (2000), https://doi.org/10.1007/3-540-44469-6_25
36. OMG: Object Management Group, Business Process Model and Notation (BPMN) Version 2.0. OMG Standard (2011)
37. OMG: Object Management Group, Unified Modeling Language (UML) Version 2.5.1. OMG Standard (2017)
38. Oriol, X., De Giacomo, G., Estañol, M., Teniente, E.: Embedding reactive behaviour into artifact-centric business process models. Future Generation of Computer Systems p. Accepted for publication (2021)
39. Park, J., Sandhu, R.: The UCON ABC usage control model. ACM Transactions on Information and System Security 7(1), 128–174 (feb 2004)
40. Pérez-Álvarez, J.M., Gómez-López, M.T., Eshuis, R., Montali, M., Gasca, R.M.: Verifying the manipulation of data objects according to business process and data models. Knowledge and Information Systems (Jan 2020), https://doi.org/10.1007/s10115-019-01431-5

41. Poels, Geert and García, Félix and Ruiz, Francisco and Piattini, Mario: Architecting business process maps. COMPUTER SCIENCE AND INFORMATION SYSTEMS 17(1), 117–139 (2020), http://dx.doi.org/10.2298/csis181118018p
42. Pozo, S., Varela-Vaca, Á.J., Gasca, R.M.: Mda-based framework for automatic generation of consistent firewall acls with NAT. In: Computational Science and Its Applications - ICCSA 2009, International Conference, Seoul, Korea, June 29-July 2, 2009, Proceedings, Part II. pp. 130–144 (2009)
43. Pérez-Álvarez, J.M., Parody, L.P., Gómez-López, M.T., Gasca, R.M., Ceravolo, P.: Decision-making support for input data in business processes according to former instances. Comput. Sci. Inf. Syst. 18(3), 597–618 (2021), https://doi.org/10.2298/CSIS200522051P
44. Queralt, A., Teniente, E.: Verification and validation of UML conceptual schemas with OCL constraints. ACM Trans. Softw. Eng. Methodol. 21(2), 13:1–13:41 (2012)
45. Rodríguez, A., Fernández-Medina, E., Trujillo, J., Piattini, M.: Secure business process model specification through a UML 2.0 activity diagram profile. Decision Support Systems 51(3), 446–465 (2011), http://dx.doi.org/10.1016/j.dss.2011.01.018
46. Rull, G., Farré, C., Queralt, A., Teniente, E., Urpí, T.: Aurus: explaining the validation of UML/OCL conceptual schemas. Softw. Syst. Model. 14(2), 953–980 (2015)
47. Rull, G., Farré, C., Teniente, E., Urpí, T.: Providing explanations for database schema validation. In: Bhowmick, S.S., Küng, J., Wagner, R.R. (eds.) DEXA 2008. LNCS, vol. 5181, pp. 660–667. Springer (2008)
48. Salnitri, M., Brucker, A.D., Giorgini, P.: From Secure Business Process Models to Secure Artifact-Centric Specifications. In: Enterprise Business-Process and Information Systems Modeling, pp. 246–262. Springer Science + Business Media (2015)
49. Salnitri, M., Dalpiaz, F., Giorgini, P.: Designing secure business processes with secbpmn. Software and System Modeling 16(3), 737–757 (2017)
50. Varela-Vaca, Á.J., Borrego, D., Gómez-López, M.T., Gasca, R.M.: A usage control model extension for the verification of security policies in artifact-centric business process models. In: BIS 2016. pp. 289–301 (2016)
51. Varela-Vaca, A.J., Galindo, J.A., Ramos-Gutiérrez, B., Gómez-López, M.T., Benavides, D.: Process Mining to Unleash Variability Management:Discovering Configuration Workflows Using Logs. In: Proceeedings of the 23nd International Systems and Software Product Line Conference, SPLC 2019, Paris, France, September 9-13, 2019. pp. – (2019), https://doi.org/10.1145/3336294.3336303
52. Varela-Vaca, Á.J., Gómez-López, M.T.: Access control security policies DSL for BPMN. http://www.idea.us.es/securitydsl/ (2020)
53. Varela-Vaca, A.J., Parody, L., Gasca, R.M., López, M.T.G.: Automatic verification and diagnosis of security risk assessments in business process models. IEEE Access 7, 26448–26465 (2019), https://doi.org/10.1109/ACCESS.2019.2901408
54. Varela-Vaca, A.J., Gasca, R.M., Ceballos, R., Gómez-López, M.T., Bernáldez Torres, P.: CyberSPL: A framework for the verification of cybersecurity policy compliance of system configurations using software product lines. Applied Sciences 9(24) (2019), https://www.mdpi.com/2076-3417/9/24/5364
55. Wainer, J., Barthelmess, P., Kumar, A.: W-RBAC — a workflow security model incorporating controlled overriding of constraints. International Journal of Cooperative Information Systems 12(04), 455–485 (Dec 2003), https://doi.org/10.1142/s0218843003000814
56. Weber, I., Hoffmann, J., Mendling, J.: Beyond soundness: on the verification of semantic business process models. Distributed Parallel Databases 27(3), 271–343 (2010)
57. Wolter, C., Menzel, M., Schaad, A., Miseldine, P., Meinel, C.: Model-driven business process security requirement specification. Journal of Systems Architecture 55(4), 211–223 (apr 2009)
58. Wolter, C., Schaad, A.: Modeling of task-based authorization constraints in BPMN. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) Business Process Management, 5th International Conference, BPM 2007, Brisbane, Australia, September 24-28, 2007, Proceedings. Lecture Notes in

Computer Science, vol. 4714, pp. 64–79. Springer (2007), https://doi.org/10.1007/978-3-540-75183-0_5
59. Zedan, H., Al-Sultan, S.: The Specification and Design of Secure Context-Aware Workflows . Expert Systems With Applications 86, 367–384 (2017)
60. Zoet, M., Versendaal, J., Ravesteyn, P.: A business rules viewpoint on risk and compliance management. In: 24th Bled eConference: eFuture Creating Solutions for the Individual, Organisations and Society, Bled, Slovenia, June 12-15, 2011. p. 25 (2011), http://aisel.aisnet.org/bled2011/25

Montserrat Estañol obtained her Ph.D. in 2016, at Universitat Politècnica de Catalunya (UPC), where she currently teaches an undergraduate course on Software Engineering. She has worked as a postdoc researcher at InLab FIB, UPC, and at Barcelona Supercomputing Center (BSC). Her research interests include conceptual modeling and ontologies, artifact-centric business process modeling and automated reasoning on both conceptual schemas and artifact-centric business process models.

Ángel Jesús Varela-Vaca received an MSc in Software Engineering and Technology (2009) and obtained his PhD with honours at the University of Seville (2013). He is currently working as Associate Professor at the Universidad Sevilla and belongs to the IDEA Research Group. He has led various private and public research projects and he has published several papers in high-impact factor journals, including Computers in Industry, ACM Computing Surveys, Empirical Software Engineering, Decision Support Systems, Information and Software Technology, Journal System and Software, Information Systems, among others. He was nominated as a member of Program Committees in different conferences, ISD 2016, BPM Workshops 2017, SIMPDA 2018, SIMPDA 2019, SPLC 2019, and SPLC 2020. He has served as a reviewer for many reputed journals.

María Teresa Gómez-López is PhD in Computer Science, Lecturer at the University of Seville and the head of the IDEA Research Group. Her research areas include Business Processes and Data management in Big Data environment. She has led several private and public research projects and has published more than twenty impact papers (DSS, IS, DKE, IST ·). She was nominated as a member of several Program Committees (BPM, ER, EDOC, CAiSE Doctoral Consortium, ·), and she has been reviewing for international journals. She has been invited speaker at various conferences and summers schools.

Rafael M. Gasca holds a PhD in computer science from the Universidad de Sevilla, in Spain. He is full professor since 2018. He has led the Quivir Research Group since 2000, since 2015, he has been a member of the IDEA Research Group at the Universidad de Sevilla. He has been the leader of different public and private research projects and has directed twelve PhD theses. He has published tens dozens of papers in high-impact factor, including IEEE Computing, IEEE Communications Magazine, Information and Software Technology, Journal System and Software, Information Systems, Information and Software Technology, and Data and Knowledge Engineering. He has been a reviewer in relevant security conferences and journals and an organiser of artificial intelligence conferences and an international summer school on fault diagnosis of complex systems.

Ernest Teniente is Professor of Software Engineering in the Department of Service and Information System Engineering at the Universitat Politècnica de Catalunya (UPC). He is also Director of inLab FIB, the innovation laboratory of the Computer Science Faculty of Barcelona, and head of the Information Modeling and Processing (IMP) research group at the UPC. His research interests include ontologies and conceptual modeling, business process management, automated reasoning, automatic code generation, integrity constraints enforcement, and data integration.

Received: February 17, 2021; Accepted: October 20, 2021.

Computer Science and Information Systems 19(2):573–593 https://doi.org/10.2298/CSIS210228064S

Enhancing Interactive Graph Representation Learning for Review-based Item Recommendation

Guojiang Shen, Jiajia Tan, Zhi Liu, and Xiangjie Kong *

College of Computer Science and Technology, Zhejiang University of Technology
Hangzhou 310023, China
xjkong@ieee.org

Abstract. Collaborative filtering has been successful in the recommendation systems of various scenarios, but it is also hampered by issues such as cold start and data sparsity. To alleviate the above problems, recent studies have attempted to integrate review information into models to improve accuracy of rating prediction. While most of the existing models respectively utilize independent module to extract the latent feature representation of user reviews and item reviews, ignoring the correlation between the latent features, which may fail to capture the similarity of user preferences and item attributes hidden in different review text. On the other hand, the graph neural network can realize the information interaction in high dimensional space through deep architecture, which has been extensively studied in many fields. Therefore, in order to explore the high dimensional relevance between users and items hidden in the review information, we propose a new recommendation model enhancing interactive graph representation learning for review-based item recommendation, named IGRec. Specifically, we construct the user-review-item graph with users/items as nodes and reviews as edges. We further add the connection of the user-user and the item-item to the graph by meta-path of user-item-user and item-user-item. Then we utilize the attention mechanism to fuse edges information into nodes and apply the multilayer graph convolutional network to learn the high-order interactive information of nodes. Finally, we obtain the final embedding of user/item and adopt the factorization machine to complete the rating prediction. Experiments on the five real-world datasets demonstrate that the proposed IGRec outperforms the state-of-the-art baselines.

Keywords: Collaborative Filtering, Recommendation, Graph Convolutional Network, Embedding.

1. Introduction

With the rise of e-commerce, personalized recommendation systems are designed to provide users with personalized information services and decision supports. Excellent recommendation systems can improve the operational efficiency of e-commerce platforms. Of course, there are some researchers who make interesting recommendations in other areas [31], [39], [36], [13]. Collaborative Filtering (CF) [16], [2], [14], [24] has been successful in the recommendation systems of various scenarios. Many of the successful CF techniques are based on Matrix Factorization (MF) [8], [23], [28] that decomposes the

* The corresponding author

user-item rating matrix into two low-dimensional matrices which represent the latent features of the user and the item respectively. However, the recommendation performance of MF methods will degrade significantly when the rating matrix is extremely sparse, and the rating only shows the overall feeling of the user, but do not explain why the user prefers to buy this item. Recently, user-generated reviews after purchasing have become a novel source of recommendation data and some works have introduced reviews to alleviate the above problems [29], [41], [1]. In these works, the convolutional neural network (CNN) architecture instead of topic models [19] is employed to extract the user and item latent features from the corresponding reviews respectively. Compared to a model that only uses rating information to make recommendations, the addition of reviews not only improve the performance of the model, but also increase its interpretability. Although these studies have improved the accuracy of the predictions, there are still some problems to be solved.

- Most of the existing works have learned the latent features of users and items in a static and independent way, which ignores the semantic relevance hidden in the review text.
- Some works have attempted to explore interactions between latent features [18], [2]. However, they hardly extract more complex and higher-order interactions information, which only utilize operation in low-dimensional space.

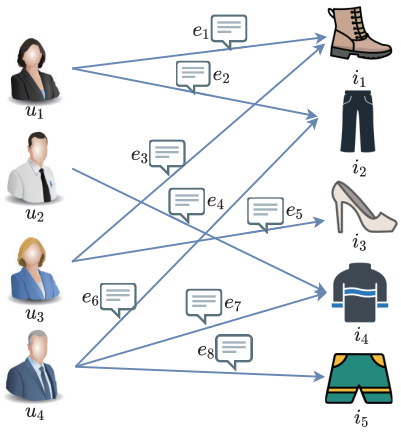


Fig. 1. The Example of User-Review-Item Graph, u_j, e_j, i_j represent user nodes, reviews, item nodes respectively.

To solve the above problems, we propose an IGRec model. The model utilizes the graph neural network to fuse edge (review) and node (user and item) information to achieve predictive rating. The contributions of this paper are summarized as follows:

- To the best of our knowledge, we are the first to introduce graph neural network to learn review-based user/item representation, and we define graph convolution operator to aggregate edges and nodes features.

- We propose a novel idea that using graph to cover the users, items, and reviews information as shown in the Figure 1 and further add the connection of the user-user and the item-item to graph. To avoid the loss of information, we extract interactive information in the whole graph.
- The experiments are performed on five real-world datasets and the experimental results show that the proposed IGRec model achieves better rating prediction accuracy than the existing state-of-the-art methods.

The remainder of this paper is organized as follows: Section 2 reviews the related work in the recommender systems. Section 3 presents the preliminaries. Section 4 introduces the overall framework of IGRec model in detail. The experimental settings and results are presented in Section 5. Section 6 concludes the article.

2. Related Work

Our work is related to two lines of literatures, the review text used for recommendation and graph neural networks for recommendation. We review the recent advances in both areas.

2.1. Review Text for Recommendation

In order to alleviate the problem of sparse data and cold-start in the recommendation system, researchers began to try to introduce auxiliary information closely related to users and items when build the model, especially, review text of users has become a research hotspot to improve the performance of recommender systems. Some researchers [17] introduced the topic model into the framework and used the review text to improve prediction accuracy and some interpretable work on the model. McAuley. et al. in [19] utilized Latent Dirichlet Allocation (LDA) to extract the topic of the reviews and couple the latent topics and ratings, which accuracy significantly improved in the task of rating prediction. The EFM [40] model and the TriRank [6] model regarded the latent topics as aspects of user and item and provided explanations for recommendation. However, these methods all belong to the category of Bag-of-words models which ignore word order and local context information. Hence, a lot of specific information in the form of phrases and sentences has been lost. To alleviate this limitation, several methods introduce the deep neural network into the traditional MF framework. Kim. et al. in [8] utilized a CNN network to obtain semantic representation of reviews and take into account the word order and local context information. CNN and probability matrix decomposition (PMF) were combined to predict the rating. Zheng. et al. in [41] used two long documents formed by concatenating all reviews of users and items as datasets and learning the representation by two convolution structures vector of the document. Finally, the embedding is concatenated and input into FM for prediction rating. Despite these models having significant improvements in recommendation performance, these works have learned the latent features of users and items in a static and independent way which neglects the information-rich interactions between users and items. Recently, attention mechanisms are fused into the model to capture the importance of different latent features [1], [34], [25] and learn user-item interactions [18]. Seo, S. et al. in [25] introduced word-level attention on the DeepCoNN, different words

are of different importance to the modeling user and the item. Chen. et al. in [1] used the attention mechanism to compute the usefulness of reviews for recommendation. Tay. et al. in [29] employed review-level co-attention to pick out the important reviews, then selected the words in the important reviews for word-level co-attention. Wu. et al. in [34] derived a joint representation for a given user-item pair based on their individual latent features and latent feature interactions. The DAML model learned user-item interactions by a dual attention mechanism that the local attention layer focuses on the importance of different words in the sentences, while the mutual attention layer focuses on the learning of feature interactions [18]. Z Wang. et al. in [33] proposed a hybrid deep collaborative filtering model that two attention-based GRU networks attempt to learn context-aware representation as textural feature for users and items from reviews. The above approach explores the importance of words in sentences, the importance of individual review to the overall document, and the importance of reviews to users and items. However, few studies above focused on the interactive importance of user-item topology graph which contain a wealth of semantic information. In this paper, we regard users and items as nodes and reviews as edges, and construct the user-item-review graph. We further add the connection of the user-user and the item-item to the graph by meta-path of user-item-user and item-user-item.

2.2. Graph Learning for Recommendation

Another direction of research exploits the user-item interaction graph to infer user preference. Yang. et al. in [38] utilized a random walk to capture higher-order relationships between users and items, combining with the degree of vertex, the positive samples of different order are sampled with certain probability, and the attenuation coefficient is assigned to the positive samples of different order. Kong X. et al. in [12] built the weighted-citation graph and the random walks were used for top-K paper recommendation. However, in recent years, there has been increasing interest in developing graph algorithms based on deep learning. The most widely used algorithm is based on Graph Convolutional Network (GCN) [11]. GCN achieved significant improvements compared to previous graph-mining methods such as DeepWalk [21]. Xue, G. et al. in [37] made a comprehensive summary of network representation technology. After that, instead of transductive learning of GCN, Hamilton. et al. in [5] proposed an inductive framework that leverages node sampling and feature aggregation function to generate node embeddings for unseen data. The GAT [30] model introduced the attention mechanism into GCN. It doesn't depend on the full graph structure, only on the edges and it can handle the case of a directed graph. By learning attention coefficients among nodes, GAT assigns different weights to different adjacency nodes, which make decisions to focus on the most relevant neighbors. Naturally, some studies began to introduce GCN into the recommendation system to extract the information of the user-item interaction graph. The NGCF model extracted the high-dimensional connection relation of the user-item through multiple convolution operations [32]. Shen, G. et al. in [26] proposed an unsupervised commercial district recommendation framework via embedding space clustering on graph convolution networks. Ge. et al. in [4] constructed a bipartite graph with the historical user click information and recommend news through the graph attention network [7]. The NIRec model captured interaction patterns of node pairs by different meta-paths in the heterogeneous network, then the attention mechanism is used to fuse the information obtained from different meta-paths.

Although these methods improve performance, they just use the information of the nodes and ignore the information of the edges that contains a wealth of semantic information. Hence, in this paper, we regard reviews as edges and use word2vec [20] and TextCNN [9] to extract review text information, then in each iteration, the aggregation and combination operator are used to extract the information of nodes and edges.

3. Preliminaries

3.1. TextCNN

Although Bert [3] and GPT [22] models have been widely used in the field of NLP recently, in order to maintain the running efficiency of the model, we still choose the combination of word2vec+TextCNN to extract text information. Next we will briefly introduce TextCNN. TextCNN consists of two layers: embedding layer, convolutional layer. In the embedding layer, a word is converted to a d dimensional vector by pre-trained embedding, such as trained Wikipedia corpus using word2vec. Then a fixed length L is extracted for each sentence, which intercept for longer and pad for shorter. The sentence can then be represented as a matrix $M \in R^{L\times d}$. Following the embedding layer is the convolutional layer, we can view embedding matrix as an image and convolutional neural network is used to extract features. Text convolution differs from image convolution in that it is convolved only in one direction (vertical) of the text sequence. Different features are extracted by different filter $K \in R^{t\times d}$. t is the sliding window length, if convolutional layer have m filters, i^{th} filter produces features as:

$$c_i = ReLU(M * K_i + b_i) \tag{1}$$

where $ReLU$ is a nonlinear activation function. $*$ is the convolution operation, b_i is the bias. After that, $c_1, c_2, \cdots {c_i}^{(T-t+1)}$ produced by i^{th} filters. Then max-pooling operation to unify features arising from different filters. which is defined as:

$$o_i = \max(c_1, c_2, \cdots {c_i}^{(T-t+1)}) \tag{2}$$

The final output of the convolutional layer is the concatenation of the output from m filters,

$$O = [o_1, o_2, \cdots o_m] \tag{3}$$

In general, the output O are passed to a fully connected layer with weight matrix W and bias g, which is:

$$X = WO + g \tag{4}$$

3.2. GCN-based Models

The researchers created the GCN so as to use convolution operations on graph-structured data. Let a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with node $v \in \mathcal{V}$, edge $(v, v') \in \mathcal{E}$. $H^0 \in R^{n\times d_0}$ represents the initialization node feature matrix, which n is the number of nodes and d_0 denotes the feature dimension of node. $H^l \in R^{n\times d_l}$ represents l-th layer hidden state of nodes, where d_l denotes the feature dimension of l-th layer node.

The original GCN [11] model following layer-wise propagation rule:

$$H^{(l+1)} = \sigma\left(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\right) \tag{5}$$

where $\tilde{A} = A + I_N$ is the added self-connections adjacency matrix of the graph $\mathcal{G}$. $\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ denotes degree matrix and $W^{(l)}$ is l-th layer trainable weight matrix. $\sigma(\cdot)$ denotes an activation function. At each propagation, the nodes are updated simultaneously. Recently, there have been many variants based on GCN. As summarized in [35], [15], a propagation layer can be separated into two sub-layers: aggregation and combination, which is defined as:

$$h_{agg}^{(l)} = \sigma\left(W^{(l)} \cdot \mathrm{AGG}\left(\left\{h_{v'}^{(l-1)}, \forall v' \in A(v)\right\}\right)\right) \tag{6}$$

$$h_v^{(l)} = \mathrm{COMBINE}\left(h_v^{(l-1)}, h_{agg}^{(l)}\right) \tag{7}$$

Here, $A(v)$ is a set of nodes adjacent to node v. $\mathrm{AGG}(\cdot)$ is a aggregation function that aggregate features from neighbors of node v. Some operators are selectable as aggregation function, such as mean-pooling, max-pooling [5] or attention mechanism [30]. $W^{(l)}$ is l-th layer trainable weight matrix. $\mathrm{AGG}(\cdot)$ denotes aggregated neighbors feature vector of node v at l-th layer. $\mathrm{COMBINE}(\cdot)$ is combination function that combine node v self feature and aggregated neighbors feature, which optional operators include element-wise product, concatenation [5] and so on. In original GCN, there is no explicit combination step, which is because the adjacency matrix in the original GCN has self-connections. Hence in aggregation step that node self feature has been combinated with those of its neighbors features.

4. The proposed model

In this section, we will introduce our IGRec model in detail. IGRec is devised to predict a rating for a new user-item pair by exploiting existing review data and user-item interaction information. As demonstrated in Figure 2, there are three components in the IGRec: (1) Eembedding layer that extract the text information for the review as edge embedding and offer ID embedding of user and item as node embedding. (2) Interaction layer that update the embedding by learning high-order connectivity relations. (3) Prediction layer that the factorization machine model is designed for final rating prediction.

4.1. Constructing Graph Model

We construct the user-item-review graph that regard users and items as nodes and reviews as edges, and further add the edges of the user-user and the item-item to the graph by meta-path of user-item-user and item-user-item.

4.2. Embedding Layer

The embedding layer initializes the embedding of nodes and edges. We model users and items via an embedding matrix in which the user and item embedding vectors have the

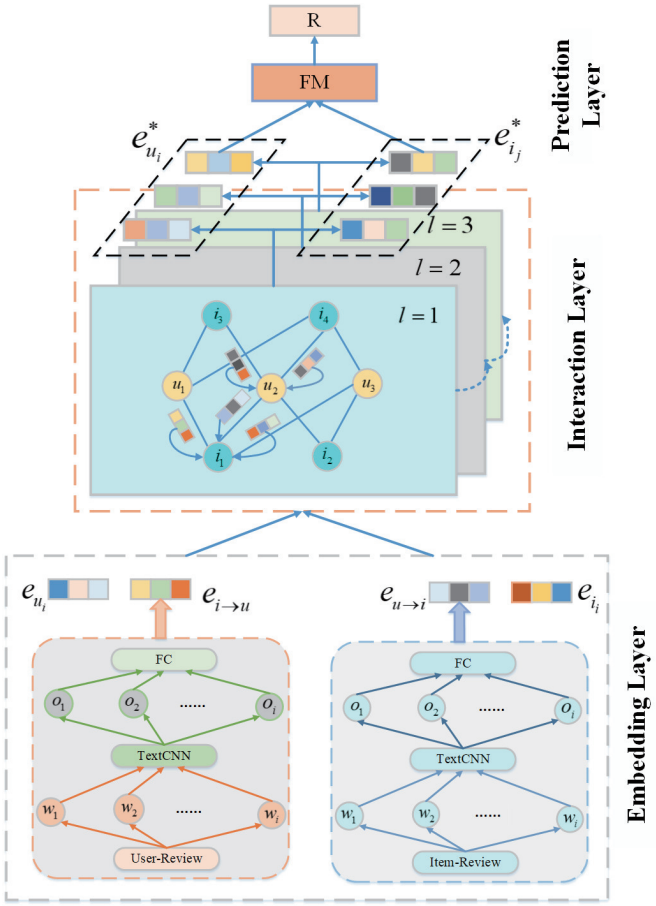


Fig. 2. The architecture of the IGRec. e_{u_i} and e_{i_j} denote initial user ID feature and item ID feature respectively. $e^*_{u_i}$ and $e^*_{i_j}$ denote final user and item feature respectively. $e_{u\to i}$ and $e_{i\to u}$ denote review feature of user and item.

same dimension d. In addition, we use word2vec to initialize the embedded representation of all reviews.

Node Embedding: Let $U = \{u_1, u_2, \cdots, u_M\}$, $I = \{i_1, i_2, \cdots, i_N\}$ denote the user set of M users and the item set of N items respectively. We design an user u (item i) with an embedding vector $e_{u_i} \in R^d$ ($e_{i_j} \in R^d$), of which user and item embedding dimension are both d. Hence, the users and items representation matrix can be defined as:

$$E_u = [e_{u_1}, e_{u_2}, \cdots, e_{u_M}] \tag{8}$$

$$E_i = [e_{i_1}, e_{i_2}, \cdots, e_{i_N}] \tag{9}$$

Since the user and item representation dimensions are equal, we can represent the nodes representation matrix that the stack of embedding matrix of users and items, which is:

$$E_n = \begin{pmatrix} E_u \\ E_i \end{pmatrix} \tag{10}$$

Here, $E_n \in R^{(M+N)\times d}$ is nodes initial representation matrix.

Edge Embedding: Let a review text $x \in X$, which contains fixed-length l words, we use word2vec model to construct the initial word vector and map each word x into the word vector, then concatenate these words vector denote review embedding matrix, which is:

$$D = [w_1, w_2, \cdots, w_l] \tag{11}$$

where $D \in R^{l\times p}$, l stands for the number of words in a single review, p is the embedding dimension of each word. Let user review text $x_u \in X_u$, item review text $x_i \in X_i$. Considering that reviews have different effects on users and items, we apply different TextCNN (cf Eq. (1,2,3)) to extract the information from the initial embedding matrix of the users and items, then we gain the embedding matrix of a single edge, which is:

$$e^*_{u\to i} = [o_{u_1}, o_{u_2}, \cdots, o_{u_d}] \tag{12}$$

$$e^*_{i\to u} = [o_{i_1}, o_{i_2}, \cdots, o_{i_d}] \tag{13}$$

where $e^*_{u\to i}$ is the edge from user to item, $e^*_{i\to u}$ is the edge from item to user. $o_{u_i} \in R^g$ and $o_{i_i} \in R^g$ respectively represent the features obtained from the convolution layer of different user and item TextCNN, g is the feature dimension after max-pooling layer. To facilitate subsequent operations, we use the dimension in which the ID embeding as the number of output channels for the convolution. Finally, we connect a full connection layer to convert the matrix to an one-dimensional vector to represent the embedding of a single edge, which is:

$$e_{u\to i} = W_u O_u + b_u \quad e_{i\to u} = W_i O_i + b_i \tag{14}$$

where $O_u \in R^{d\times g}$, $O_i \in R^{d\times g}$ respectively denote the concatenation of the $[o_{u_1}, o_{u_2}, \cdots, o_{u_d}]$, $[o_{i_1}, o_{i_2}, \cdots, o_{i_d}]$, $W_u, W_i \in R^{g\times d}$ are trainable weight matrices, b_u, b_i are the bias.

4.3. Interaction Layer

The interaction layer is the most important layer of the model. It mainly solves two problems. One is how to fuse edge information into the node, the other is how to transfer the node information with edge information to other nodes. For the first problem, the edge information fusion sub-layer uses the attention mechanism to allocate edge weight so as to provide greater weight values for important reviews. For the second problem, the message propagation sub-layer that is similar to GCN are introduced to fuse the features of adjacent nodes and the multi-hop node information can be obtained by repeated training of the sub-layer.

Edge Information Fusion: In order to realize the training of mini-batch, we fuse the edges information into the nodes and transform the problem into the information transmission between the nodes. Since a single node connects multiple edges, how to selectively

pick out the important edges is a problem we need to solve. In recent years, attention mechanism has been frequently introduced into the recommendation system model [1], [19], [18] and improved the performance of recommender systems. Hence, the model use a linear-layer network to compute the attention score of the edges connected to a single node. Let an user node $e_{u_i} \in R^d$, user edge $e_{u_i \to i_j} \in R^d$ which e_{u_i} is the user node embedding and $e_{u_i \to i_j}$ is the edge embedding that obtained through the embedding layer. $E_{u_i \to i} = [e_{u_i \to i_1}, e_{u_i \to i_2}, \cdots, e_{u_i \to i_m}]$ is the edge set of user node e_{u_i} and as the input of attention layer, the attention network is defined as:

$$\alpha^*_{uj} = ReLU(W_u E_{u_i \to i} + b_u) \tag{15}$$

where $W_u \in R^{d \times m}$, $b_u \in R^m$ are model parameters, $ReLU$ is a nonlinear activation function.

Softmax function was used to normalize the above attention score to obtain the final weight of the edges (reviews), which could be interpreted as the contribution of m edges to user u_i:

$$\alpha_{uj} = \frac{exp(\alpha^*_{uj})}{\sum_{j=1}^{m} exp(\alpha^*_{uj})} \tag{16}$$

After that, we obtain the attention weight of each edge, the feature vector of edges set for user node u_i is calculated as the following weighted sum:

$$e_{u_i \to i} = \sum_{j=1}^{m} \alpha_{uj} e_{u_i \to i_j} \tag{17}$$

where $e_{u_i \to i} \in R^d$ is the attention-weighted embedding sum of the connecting edges of node u_i.

For the item node, we do the same processing. Let an item embedding $e_{i_i} \in R^d$, through the attention layer, the adjacent edge information is aggregated, which is:

$$e_{i_i \to u} = \sum_{j=1}^{m} \alpha_{ij} e_{i_i \to u_j} \tag{18}$$

where $e_{i_i \to u} \in R^d$ is the attention-weighted embedding sum of the connecting edges for item node i_i.

After that, features of adjacent edges are fused into the embedding of the node, the model uses the element-wise product to combine these two kinds of embeddings, which is:

$$e^{\Delta}_{u_i} = e_{u_i} \odot e_{u_i \to i} \quad e^{\Delta}_{i_i} = e_{i_i} \odot e_{i_i \to u} \tag{19}$$

where $e^{\Delta}_{u_i} \in R^d$, $e^{\Delta}_{i_i} \in R^d$ respectively represent the user and the item embedding that have fused edges information. $\odot$ is element-wise product.

Message Propagation Layer: After the edge information fusion sub-layer, we get the node embedding that fused edge information. In this layer, the model uses aggregation and combination operators to extract interactive information of latent features in the graph.

Aggregation Operator: For a pair of connected user-item pair (u_i, i_j). $e_{u_i} \in R^d$, $e_{i_j} \in R^d$ respectively represent embedding of the user u_i and the item i_j. Since the user

and item have the same embedding dimension, we make an unified matrix representation of the nodes. $\mathcal{H}_n = [e^{\Delta}_{u_1}, \cdots e^{\Delta}_{u_M}, e^{\Delta}_{i_1}, \cdots e^{\Delta}_{u_M}]$, the aggregation process of l-th layer is defined as:

$$Agg_n^{(l)} = ReLU({W_1}^{(l)}\mathcal{L}\mathcal{H}_n^{(l-1)} + {b_1}^{(l)}) \tag{20}$$

where $\mathcal{L}$ represents the Laplacian matrix for the user-item graph, which is formulated as:

$$\mathcal{L} = I + D^{-\frac{1}{2}}AD^{-\frac{1}{2}} \quad and \quad A = \begin{bmatrix} U^* & R \\ R^T & I^* \end{bmatrix} \tag{21}$$

Instead of the traditional adjacency matrix, we further add user-user and item-item connections into adjacency matrix A. $U^* \in R^{M\times M}$ is the connections of user-user by meta-path user-item-user. $I^* \in R^{N\times N}$ is the connections of item-item by meta-path item-user-item.

R is the the user-item interaction matrix. D is the degree matrix. ${W_1}^{(l)}$ is the l-th layer trainable parameter matrix, ${b_1}^{(l)}$ is the l-th layer bias. $ReLU$ is the nonlinear activation function.

Combination Operator: After the information of the adjacent node is collected, it is necessary to fuse it with the information of the node itself, which is formulated as:

$$Comb_n^{(l)} = {W_2}^{(l)}(\mathcal{H}_n^{(l-1)} \odot Agg_n^{(l)}) + {b_2}^{(l)} \tag{22}$$

where $\odot$ denotes the element-wise product that information can be transmitted through the correlation between nodes. ${W_2}^{(l)}$ denotes the importance of the adjacent node, and ${b_2}^{(l)}$ is the l-th layer bias.

4.4. Prediction Layer

After L-layer convolution (aggregation and combination) operations, for user node u_i, we obtain multiple representations $\{e^{\Delta^{(0)}}_{u_i}, \cdots, e^{\Delta^{(L)}}_{u_i}\}$. Just like the two dimensional convolution, different convolutions may acquire different latent features. We concatenate them to constitute the final embedding for the user. In addition, for an item i_j, we do the same operation to concatenate item embedding $\{e^{\Delta^{(0)}}_{i_j}, \cdots, e^{\Delta^{(L)}}_{i_j}\}$ and get the final item embedding:

$$e^*_{u_i} = e^{\Delta^{(0)}}_{u_i}||\cdots||e^{\Delta^{(L)}}_{u_i} \quad and \quad e^*_{i_j} = e^{\Delta^{(0)}}_{i_j}||\cdots||e^{\Delta^{(L)}}_{i_j} \tag{23}$$

The concatenation of $[e^*_{u_i}, e^*_{i_j}]$ is passed into a factorization machine (FM). The FM function is defined as follows:

$$t2F(x) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n}\sum_{j=i+1}^{n} \langle v_i, v_j\rangle x_i x_j \tag{24}$$

where $x \in R^k$ is the input feature vector. $\langle ., .\rangle$ is the element-wise product. The parameters v_i are factorized parameters used to model pairwise interactions (x_i, x_j). w_0 is the bias, $\sum_{i=1}^{n} w_i x_i$ represents a linear regression. The output of FM is the final rating of user-item pair:

$$\widehat{R}_{u,i} = FM([e^*_u, e^*_i]) \tag{25}$$

where e^*_u, e^*_i respectively represent the final users and the items embedding.

4.5. Learning

Because the task of this paper is regression, we exploit squared loss as the objective function:

$$loss = \sum_{(u,i)\in\Gamma} (\widehat{R}_{u,i} - R_{u,i})^2 + \lambda_\Theta \|\Theta\|^2 \tag{26}$$

where Γ denotes the set of instances for training, $R_{u,i}$ is the ground truth rating assigned by the user u to the item i.

$\widehat{R}_{u,i}$ is the prediction rating, Θ denotes all the parameters of the model is used as regularization to prevent the model from overfitting. The entire framework can be effectively trained by using end-to-end paradigm reverse propagation.

To optimize the objective function, we adopt the Adaptive Moment Estimation (Adam) [10] as the optimizer. Additionally, to prevent overfitting, we adopt dropout strategy [27] to the linear layer of the model.

Table 1. Statistics of datasets used in this paper

Dataset	users	items	ratings	reviews per user	reviews per item	length of review	density
Office Products	4905	2420	53258	14	35	124	0.44%
Amazon Instant Video	5130	1685	37126	8	27	101	0.43%
Toys and Games	2555	2211	19925	8	11	117	0.35%
Digital Music	5541	3568	64706	13	24	202	0.32%
Beauty	15201	9680	154150	11	19	97	0.11%

5. Experiments

In this section, we present our experimental setup and empirical evaluation. Our experiments are designed to answer the following research questions (RQs):

(1) **RQ1**-How does IGRec perform as compared with state-of-the-art review-based recommendation methods?

(2) **RQ2**-How do different hyper-parameter settings, such as depth of 'GCN' layer and mode of information transmission of interaction layer, affect IGRec?

(3) **RQ3**-Does the model really take advantage of the review information, and what is the effect of removing the review information?

5.1. Datasets

In our experiments, we used five publicly accessible datasets to evaluate our model. The five datasets are from Amazon 5-core, which include reviews on Office Products, Amazon Instant Video, Toys and Games, Digital Music, and Beauty. Since the raw data is very large and sparse, we use data preprocessing to ensure that each user and item has at least one review. For each dataset, we selected a fixed number of reviews for users and items

respectively. For each review, we selected a fixed-length word for TextCNN to extract the text information, which intercept for longer and pad for shorter. The model selects value at three-quarters of the total review length value as the hyper-parameter. The characteristics of these datasets are shown in Table 1.

In the experiments, we randomly split each dataset into three parts: training set (80 %), validation set (10 %) and test set (10 %). The final performance comparison results derive from the test set.

5.2. Evaluation Metric

The Mean Square Error (MSE) is adopted for performance evaluation:

$$MSE = \frac{1}{T} \sum_{(u,i) \in T} (\widehat{R}_{u,i} - R_{u,i})^2 \tag{27}$$

where T is the set of the user-item pairs in the testing set.

5.3. Baselines

To verify the performance of the IGRec model proposed in this paper, we compared the model with the following state-of-art recommendation methods.

- **DeepCoNN**[41]: Deep collaborative neural network is based on two parallel CNNs to learn the latent feature vectors of user and item from user review documents and item review documents respectively, and FM is used for rating prediction.
- **D_ATTN**[25]: Dual attention mechanisms that local and global attention are used to achieve the interpretability of latent features of user and item.
- **NARRE**[1]: Neural attentional regression model exploits two parallel CNNs and attention mechanism to learn the latent features of reviews, and integrates the reviews and items to complete the rating prediction.
- **NGCF**[32]: Neural graph collaborative filtering model only uses interactive connection data to extract high-dimensional interactive information by using the graph neural network, and then makes recommendations based on the learned embedding vectors of users and items.
- **MPCN**[29]: Multi-pointer co-attention networks utilize review-level co-attention and word-level co-attention by multi-pointer-learning to gain latent features of reviews, the final predicted rating is obtained through FM.
- **DAML**[18]: Dual attention mutual learning model exploits the local attention and mutual attention to learn latent features and integrate the rating and review features into an unified neural network to predict rating.

5.4. Parameter Setting

We use grid search to tune the hyper-parameters for all the methods based on the setting strategies reported by their papers. The latent dimension size is optimized from [8, 16, 32, 64, 128]. The embedding dimension size of the word in all models is set to 300. We set the batchsize to 128 for all models. The learning rate is tuned from [0.01, 0.001].

The range of dropout ratio is searched in [0.1, 0.3, 0.5, 0.7]. For the CNN text training module, the number of convolution filters is set to 100, the size of the sliding window is 3. The regularization parameter λ_Θ is set to 0.001. FM is used as the prediction layer for all models, which v is set to 10. For the NGCF model, the number of GCN layers is fine-tuned in [1, 2, 3, 4].

For IGRec, the dimensions of user embedding and item embedding are set to 8, embedding dimension of the word is set to 300, the sliding window size is set to 3, the dropout ratio is searched in [0.1, 0.3, 0.5], the learning rate is set to 0.001, the regularization parameter λ_Θ is set to 0.001, the depth of the 'GCN' layer is set to 2. v is set to 10 for FM rating prediction.

Table 2. Performance comparison on five datasets for all methods. The best and the second best results are highlighted by boldface and underlined respectively. $\Delta\%$ denotes the improvement of IGRec over the best baseline performer.

Method	Office Products	Amazon Instant Video	Toys and Games	Digital Music	Beauty
DeepCoNN	0.7337	0.9634	0.9789	0.8129	1.2052
D_ATTN	0.7064	0.9663	0.8854	0.8135	1.2113
NARRE	0.6931	0.9737	0.9108	0.8098	1.2035
NGCF	0.7066	0.9930	0.8904	0.8065	1.2182
MPCN	0.7109	0.9645	<u>0.8781</u>	0.8655	1.2386
DAML	<u>0.6852</u>	<u>0.9583</u>	0.9085	<u>0.8037</u>	<u>1.1878</u>
IGRec	**0.6728**	**0.9428**	**0.8660**	**0.7798**	**1.1520**
$\Delta\%$	1.80	1.61	1.37	2.97	3.01

5.5. Performance Comparison

The overall performance of all methods is reported in Table 2. We can see that the IGRec outperforms the baselines on the five datasets. This ascertains the effectiveness of our proposed model and clearly answers **RQ1**. Here, we make the following observations.

First, we can see that DeepConn performs worst on all five datasets. However, it is not far behind the other two CNN-based methods in some datasets. Such as in the Digital Music dataset, DeepConn and D_ATTN have similar performances, in the Beauty dataset, DeepConn and NARRE gain similar MSE. This can be explained by the fact that although the model can capture reviews information to some extent, the lack of interaction information affects the effect of the model. Surprisingly, the NGCF model does not use review data, but only interactive connection data. The performance of the NGCF model was good in all five datasets, which also verified the importance of interactive connection data. Further improvement of the NGCF model was limited by the absence of review information.

Second, MPCN is not stable in the five datasets, especially in the Digital Music dataset, its performance lags far behind that of other models. However, it get the second best result in the Toys and Games dataset. One possible explanation is that MPCN can

get better extraction of reviews information through word-level co-attention and review-level co-attention. However, in the sparse data environment, the review information may be less important than the interaction information between users and items. On the other hand, the prediction performance is adversely affected by irrelevant information within the reviews. The performance of DAML can be used as evidence to explain the importance of interactive information. DAML uses dual attention mechanism to learn user-item interactions and get the second best result in the four datasets. This can be explained that DAML not only uses the local attention mechanism to learn the importance of words in sentences, but also uses mutual attention to extract the interactive information of latent features. However, the model uses an attention mechanism alone and may fail to capture the full high-order feature interaction. Third, IGRec consistently achieves the best MSE

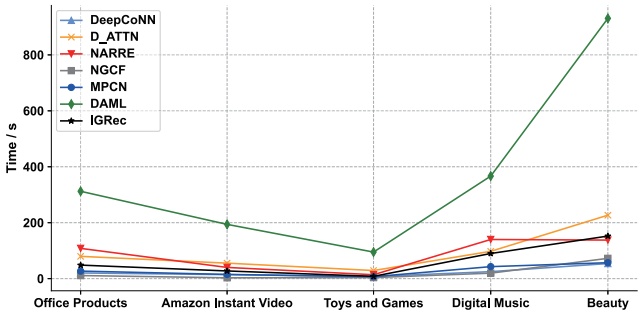


Fig. 3. The average consuming time of each epoch of different models in all five datasets.

scores across the five datasets. Surprisingly, as data sparsity increases, so does the performance of the model. Compared with the DAML model, IGRec does not use a more complex information extraction process for the review text information, but only extracts the review information through TextCNN, so the complexity of the model decreases dramatically. Compared with the review text information, we pay more attention to the interactive information extraction of latent features. The graph neural network for interactive information extraction can make up for the lack of review text information extraction, hence improving the model prediction performance.

As shown in Figure 3, we unified the batchsize of all models as 128, and calculated the average time spent for each epoch. The results show that the DAML model is the most time-consuming among all the models due to the fine-grained processing of the review text data. Because the NGCF model does not process review data, it can get training results more quickly when the number of users and items is small. However, when the number of users and items is large, for example in Beauty, it consumes more time than DeepCoNN and MPCN, which is caused by the increase of nodes and interaction data in the graph neural network. Although our model has a slight increase in time consumption compared with NGCF, DeepCoNN and MPCN, it consumes less time compared with other models. In addition, compared with NGCF, DeepCoNN and MPCN, the accuracy

of our model has been greatly improved, which shows that the efficiency of our model is the best.

5.6. Parameter Sensitivity Analysis

In order to answer **RQ2**, we mainly analyze: the dimension of latent feature, dropout ratio, depth of 'GCN' layer and mode of information transmission of interaction layer.

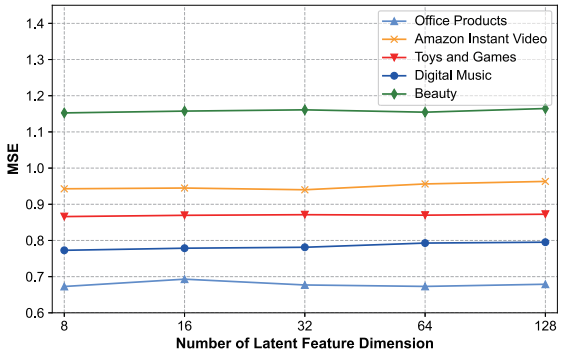


Fig. 4. The plots of the impact of latent feature dimension number.

The Dimension of Latent Feature: As shown in Figure 4, the dimension of latent feature is tuned from [8, 16, 32, 64, 128]. We can find that the performance of the IGRec model changes very little as the dimensions change. Even if the dimension number is much smaller, such as 8, the model can still achieve nearly optimal prediction accuracy in the five datasets. So we believe the latent feature dimension have little effect on the experimental performance. Moreover, the increase of latent feature dimension can increase the computational complexity of the model. Therefore, the dimension of the latent feature is set to 8.

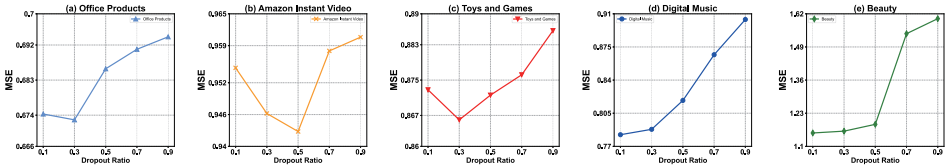


Fig. 5. The plots of the impact of dropout ratio.

The Impact of Dropout: As shown in Figure 5, a suitable dropout ratio greatly affects the performance of the model, which tends to perform better in smaller cases. The best dropout ratio is different for the five datasets. For Amazon Instant Video dataset, the best

dropout ratio is 0.5. While the best dropout for other datasets is 0.1 or 0.3. So the dropout ratio for the model is searched in [0.1, 0.3, 0.5].

Table 3. Performance comparison on five datasets for depth of the 'GCN' layer. The best results are highlighted by boldface.

	Office Products	Amazon Instant Video	Toys and Games	Digital Music	Beauty
l=0	0.6856	0.9612	0.8836	0.8392	1.2334
l=1	0.6812	0.9536	0.8748	0.8021	1.1845
l=2	0.6728	**0.9428**	**0.8660**	0.7798	**1.1525**
l=3	**0.6723**	0.9468	0.8717	**0.7732**	1.1532
l=4	0.6842	0.9568	0.8730	0.8045	1.1872

Table 4. Performance comparison on five datasets for mode of the 'GCN' layer. The best results are highlighted by boldface.

	Office Products	Amazon Instant Video	Toys and Games	Digital Music	Beauty
$Contact + Linear$	0.7023	0.9832	0.9089	0.8129	1.2052
$Dot + Attention$	0.6934	0.9726	0.8854	0.8135	1.1913
$Dot + Linear$	**0.6812**	**0.9536**	**0.8748**	**0.8021**	**1.1845**

Depth of 'GCN' Layer: To investigate whether IGRec can benefit from multiple 'GCN' layers in the interaction layer, we vary the model depth, which search the layer numbers in the range of [0, 1, 2, 3, 4]. Table 3 summarizes the experimental results, l is the number of 'GCN' layers. Through experiments, we have the following observations. As the number of layers increases, the performance of the model will improve at first. On some datasets (Amazon Instant Video, Toys and Games, Beauty), the two layers are optimal, while on other datasets (Office Products, Digital Music), the l=3 are optimal. Different from the NGCF model in the original paper, which the model experimental results show optimal number obtained when l=3 or l=4. In this paper, we found through experiments that the NGCF model and our model often reached the optimum at l=2. This is because we add the connection of user-user and the connection of item-item so that the user and item information can be included at one hop of the adjacency node. Therefore, the adjacency matrix that we designed to enable the model to aggregate information more quickly, and reduce the calculation of the model. When the depth of layer exceeds three, the performance of the model begins to degrade. This may be due to that deeper structure may cause noise for embedding. On the other hand, when l=0, it means that 'GCN' is not used for relation extraction, and the performance on all data sets is the worst, which verifies the effectiveness of 'GCN'.

Mode of Information Transmission: To investigate how the 'GCN' layer affects the performance, we transform the different message propagation functions in the interaction layer and set depth of 'GCN' layer l=1. We fixed the aggregation operator, only changed the combination operator. Table 4 summarizes the experimental results. $Linear$ denotes the full connection layer, Dot is the element-wise product, $Attention$ is the attention mechanism that defined as:

$$u_i = tanh(Wh_i + b) \tag{28}$$

$$a_i = \frac{exp(u_i^{\mathrm{T}} u_w)}{\sum_i exp(u_i^{\mathrm{T}} u_w)} \tag{29}$$

$$s = \sum_i a_i h_i \tag{30}$$

In Table 4, we can see that the combination of $Dot + Linear$ achieves the best performance in the all five datasets. The reason for the poor performance of $Contact + Linear$ may be that in each mini-batch training, the $Contact$ operation is to contact the information of all nodes. However, each mini-batch only contains the information of relevant nodes, the model must global updates in each mini-batch iteration, which lead to noise and affect the performance. The same limitation apply to attention mechanism, it is the calculation of global attention in each mini-batch iteration, all nodes are involved in the calculation, and some irrelevant nodes affect the calculation of attention. However, for the combination of $Dot + Linear$, in each batch of training, each embedding only interacts with the embedding at the corresponding position, which avoids unnecessary noise and improves model performance.

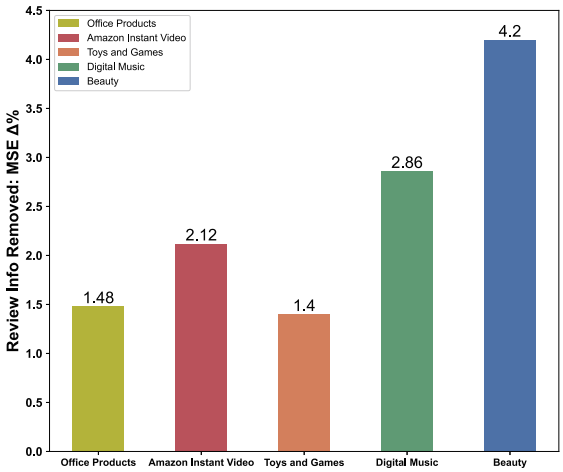


Fig. 6. The plots of the impact of review information. $\Delta\%$ denotes the percentage of performance degradation after the deletion of review information.

5.7. The Review Importance Analysis

To answer **RQ3**, we conducted an experiment in which review information was removed from five datasets. The experimental results are shown in Figure 6. We can see that in the five datasets, after the review information is removed, the performance of the model is significantly reduced, which indicates that the review information is an important resource for the recommendation system. Particularly in the Beauty dataset, the deletion of review information resulted in the greatest degradation of model performance. This may be because Beauty is the sparsest of the five datasets, the review information as additional information can greatly compensate for the lack of interaction information and thus improve the performance of the model.

6. Conclusion

In this work, we propose a novel enhancing interactive graph representation learning for review-based item recommendation. In this model, we construct the graph with users and items as nodes and reviews as edges, and further add the connection of the user-user and the item-item to the graph. TextCNN is used to extract review text information as edge embedding. The attention mechanism is developed to fuse edges information into node and the graph neural network is exploited for high-dimensional information interaction of nodes. Finally, The learned embedding of user-item pair is inputted in factorization machine to get the final rating. Experiments on five real-world datasets from Amazon show that our method consistently outperforms the existing state-of-the-art methods.

Acknowledgments. This work is partially supported by the National Natural Science Foundation of China (62073295, 62072409), Zhejiang Provincial Natural Science Foundation (LR21F020003), and Fundamental Research Funds for the Provincial Universities of Zhejiang (RF-B2020001). Zhejiang Province Basic Public Welfare Research Project under Grant No. LGG20F030008.

References

1. Chen, C., Zhang, M., Liu, Y., Ma, S.: Neural attentional rating regression with review-level explanations. In: Proceedings of the 2018 World Wide Web Conference. pp. 1583–1592 (2018)
2. Deshpande, M., Karypis, G.: Item-based top-n recommendation algorithms. ACM Transactions on Information Systems (TOIS) 22(1), 143–177 (2004)
3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
4. Ge, S., Wu, C., Wu, F., Qi, T., Huang, Y.: Graph enhanced representation learning for news recommendation. In: Proceedings of The Web Conference 2020. pp. 2863–2869 (2020)
5. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: Advances in Neural Information Processing Systems. pp. 1024–1034 (2017)
6. He, X., Chen, T., Kan, M.Y., Chen, X.: Trirank: Review-aware explainable recommendation by modeling aspects. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. pp. 1661–1670 (2015)
7. Jin, J., Qin, J., Fang, Y., Du, K., Zhang, W., Yu, Y., Zhang, Z., Smola, A.J.: An efficient neighborhood-based interaction model for recommendation on heterogeneous graph. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 75–84 (2020)

8. Kim, D., Park, C., Oh, J., Lee, S., Yu, H.: Convolutional matrix factorization for document context-aware recommendation. In: Proceedings of the 10th ACM Conference on Recommender Systems. pp. 233–240 (2016)
9. Kim, Y.: Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882 (2014)
10. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
11. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
12. Kong, X., Mao, M., Wang, W., Liu, J., Xu, B.: Voprec: Vector representation learning of papers with text information and structural identity for recommendation. IEEE Transactions on Emerging Topics in Computing (2018)
13. Kong, X., Zhang, J., Zhang, D., Bu, Y., Ding, Y., Xia, F.: The gene of scientific success. ACM Transactions on Knowledge Discovery from Data (TKDD) 14(4), 1–19 (2020)
14. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer 42(8), 30–37 (2009)
15. Li, A., Qin, Z., Liu, R., Yang, Y., Li, D.: Spam review detection with graph convolutional networks. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management. pp. 2703–2711 (2019)
16. Linden, G., Smith, B., York, J.: Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing 7(1), 76–80 (2003)
17. Ling, G., Lyu, M.R., King, I.: Ratings meet reviews, a combined approach to recommend. In: Proceedings of the 8th ACM Conference on Recommender Systems. pp. 105–112 (2014)
18. Liu, D., Li, J., Du, B., Chang, J., Gao, R.: Daml: Dual attention mutual learning between ratings and reviews for item recommendation. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 344–352 (2019)
19. McAuley, J., Leskovec, J.: Hidden factors and hidden topics: understanding rating dimensions with review text. In: Proceedings of the 7th ACM conference on Recommender Systems. pp. 165–172 (2013)
20. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems. pp. 3111–3119 (2013)
21. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: Online learning of social representations. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 701–710 (2014)
22. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners. OpenAI Blog 1(8), 9 (2019)
23. Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. In: Recommender Systems Handbook, pp. 1–35. Springer (2011)
24. Salakhutdinov, R., Mnih, A., Hinton, G.: Restricted boltzmann machines for collaborative filtering. In: Proceedings of the 24th International Conference on Machine Learning. pp. 791–798 (2007)
25. Seo, S., Huang, J., Yang, H., Liu, Y.: Interpretable convolutional neural networks with dual local and global attention for review rating prediction. In: Proceedings of the 11th ACM Conference on Recommender Systems. pp. 297–305 (2017)
26. Shen, G., Zhao, Z., Kong, X.: Gcn2cdd: a commercial district discovery framework via embedding space clustering on graph convolution networks. IEEE Transactions on Industrial Informatics (2021)
27. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research 15(1), 1929–1958 (2014)

28. Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. Advances in Artificial Intelligence 2009 (2009)
29. Tay, Y., Luu, A.T., Hui, S.C.: Multi-pointer co-attention networks for recommendation. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 2309–2318 (2018)
30. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv preprint arXiv:1710.10903 (2017)
31. Wang, W., Liu, J., Yang, Z., Kong, X., Xia, F.: Sustainable collaborator recommendation based on conference closure. IEEE Transactions on Computational Social Systems 6(2), 311–322 (2019)
32. Wang, X., He, X., Wang, M., Feng, F., Chua, T.S.: Neural graph collaborative filtering. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 165–174 (2019)
33. Wang, Z., Xia, H., Du, B., Chen, S., Chun, G.: Joint representation learning with ratings and reviews for recommendation. Neurocomputing (2020)
34. Wu, L., Quan, C., Li, C., Wang, Q., Zheng, B., Luo, X.: A context-aware user-item representation learning for item recommendation. ACM Transactions on Information Systems (TOIS) 37(2), 1–29 (2019)
35. Xu, K., Hu, W., Leskovec, J., Jegelka, S.: How powerful are graph neural networks? arXiv preprint arXiv:1810.00826 (2018)
36. Xu, Z., Jiang, H., Kong, X., Kang, J., Wang, W., Xia, F.: Cross-domain item recommendation based on user similarity. Computer Science and Information Systems 13(2), 359–373 (2016)
37. Xue, G., Zhong, M., Li, J., Chen, J., Zhai, C., Kong, R.: Dynamic network embedding survey. arXiv preprint arXiv:2103.15447 (2021)
38. Yang, J.H., Chen, C.M., Wang, C.J., Tsai, M.F.: Hop-rec: high-order proximity for implicit recommendation. In: Proceedings of the 12th ACM Conference on Recommender Systems. pp. 140–144 (2018)
39. Załuski, A., Ganzha, M., Paprzycki, M., Bădică, C., Bădică, A., Ivanović, M., Fidanova, S., Lirkov, I.: Experimenting with facilitating collaborative travel recommendations. In: 2019 23rd International Conference on System Theory, Control and Computing (ICSTCC). pp. 260–265. IEEE (2019)
40. Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y., Ma, S.: Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In: Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval. pp. 83–92 (2014)
41. Zheng, L., Noroozi, V., Yu, P.S.: Joint deep modeling of users and items using reviews for recommendation. In: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining. pp. 425–434 (2017)

Guojiang Shen received the B.Sc. degree in control theory and control engineering and the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 1999 and 2004, respectively. He is currently a Professor with College of Computer Science and Technology, Zhejiang University of Technology. His current research interests include artificial intelligence theory, big data analytics, and intelligent transportation systems.

Jiajia Tan received the B.S. degree in Mathematics and Applied Mathematics from Xiamen University of Technology, China, in 2018. He is currently pursuing the M.S. degree in Zhejiang University of Technology. His current research interests include graph neural network and recommendation system.

Zhi Liu received the B.S. degree in Automatic Control and the M.S. degree in System Engineering from Xi'an Jiaotong University, Xi'an, China, in 1991 and 1994, respectively, and the Ph.D. degree in Computer Science and Technology from Zhejiang University, Hangzhou, China, in 2001.She is currently a Professor in the College of Computer Science and Technology, Zhejiang University of Technology. She is a member of the China Computer Federation. Her current main research interests include intelligent transportation system and intelligent computing.

Xiangjie Kong received the B.Sc. and Ph.D. degrees from Zhejiang University, Hangzhou, China. He is currently a Full Professor with College of Computer Science and Technology, Zhejiang University of Technology. Previously, he was an Associate Professor with the School of Software, Dalian University of Technology, China. He has published over 150 scientific papers in international journals and conferences (with over 130 indexed by ISI SCIE). His research interests include network science, mobile computing, and computational social science. He is a Distinguished Member of CCF, a Senior Member of the IEEE, and a Member of ACM.

Received: February 28, 2021; Accepted: November 15, 2021.

Computer Science and Information Systems 19(2):595–617 https://doi.org/ 10.2298/CSIS210410052F

A Study on Optimally Constructed Compactly Supported Orthogonal Wavelet Filters

Yongkai Fan[1,2], Qian Hu[1,2,*], Yun Pan[1,2], Chaosheng Huang[3], Chao Chen[1], Kuan-Ching Li[4 *],Weiguo Lin[1,2], Xingang Wu[3], Yaxuan Li[1,2], and Wenqian Shang[1,2]

[1] State Key Laboratory of Media Convergence and Communication, Communication University of China, 100024, Beijing, China
[2] School of Computer and Cyber Sciences, Communication University of China, 100024, Beijing, China
{fanyongkai, huqian-cuc, panyun, chenchao, weilin, li_yaxuan, shangwenqian}@cuc.edu.cn
[3] School of Vehicle and Mobility, Tsinghua University, 100085,Beijing, China
{huangchaosheng,wuxingang}@tsinghua.edu.cn
[4] Dept. of Computer Science and Information Engineering (CSIE), Providence University, 43301, Taichung, Taiwan
kuancli@pu.edu.tw

Abstract. Compactly supported orthogonal wavelet filters are extensively applied to the analysis and description of abrupt signals in fields such as multimedia. Based on the application of an elementary method for compactly supported orthogonal wavelet filters and the construction of a system of nonlinear equations for filter coefficients, we design compactly supported orthogonal wavelet filters, in which both the scaling and wavelet functions have many vanishing moments, by approximately solving the system of nonlinear equations. However, when solving such a system about filter coefficients of compactly supported wavelets, the most widely used method, the Newton Iteration method, cannot converge to the solution if the selected initial value is not near the exact solution. For such, we propose optimization algorithms for the Gauss-Newton type method that expand the selection range of initial values. The proposed method is optimal and promising when compared to other works, by analyzing the experimental results obtained in terms of accuracy, iteration times, solution speed, and complexity.

Keywords: compactly supported orthogonal wavelets, the least-squares method, Gauss-Newton methods, LMF method.

1. Introduction

In signal analysis, modeling and processing, wavelets with compact support property and orthogonality are widely used. The advantages of wavelets with compact support property are: 1) when transforming truncated signals into finite-length signals, due to the infinite time-domain waveform length of compactly supported wavelet, the transformed signals are of finite length, i.e., there are no truncation errors; 2) FIR is the

* Corresponding authors

filter of compact-supported wavelets, which can make computation decreased to get favorable real-time property. Furthermore, orthogonal wavelets are useful to extract signal features for pattern recognition when applying in signal decomposition. The orthogonality reflected in digital image processing means that the original image's total energy is equal to the wavelet domain. So constructing compactly supported orthogonal wavelets plays an essential role in wavelet analysis.

Several researches to construct compactly supported orthogonal wavelets for general purposes were proposed, due to the several application advantages in multimedia technology. In [1], a framework for designing the compactly supported orthogonal wavelets in the time-domain was proposed, while an approach to design the compactly supported orthogonal wavelets from filter banks in poly-phase also having vanishing moments was proposed in [2]. Han et al. [3] constructed a family of compactly supported symmetric orthogonal complex wavelets with dilation 4 and the shortest possible supports to their orders of vanishing moments. Hiroshi Toda et al. [4] proposed a new type of orthonormal wavelet basis having customizable frequency bands. Its frequency bands can be freely designed with arbitrary bounds in the frequency domain. In [5, 6], a series of compactly supported orthogonal wavelet bases with various features were constructed based on the structure of orthogonal conjugate filters.

There are also some works for matched compactly supported orthogonal wavelets. A design for matched wavelets and matched scaling function was proposed in [7], which was done in the time-domain and did not have assumptions on the template function. In [8], the problem of designing the compactly supported orthogonal wavelets for finite-length signals was firstly addressed. In [9], a novel approach was presented to design orthogonal wavelets matched to a signal with compact support and vanishing moments. It provided a systematic and versatile framework for matching an orthogonal wavelet to a specific signal or application.

Many researchers were interested in the study of constructing compactly supported orthogonal wavelets with B-splines. He, T. and T. Nguyen[10] gave an approach to prove Daubechies' result on the existence of spline type orthogonal scaling functions and to evaluate Daubechies scaling functions. Tung Nguyen[11] presented a method to construct orthogonal spline-type scaling functions by using B-spline functions. To induce the orthogonality and remain property of compact support, a class of polynomial function factors to the B-splines' masks was multiplied. In [12], Yang Shouzhi *et al.* used orthonormalization procedure so that splines become orthogonal scaling functions. Then to make them have the property of compact support, the weighted average method was used to eliminate the denominator of the two-scale symbol. Gupta, K.L. et al. developed a simple procedure to generate the compactly supported orthogonal scaling function for higher-order B-splines and multiplied the mask of B-spline with a polynomial function that satisfied all the conditions as the mask of B-spline to obtain orthogonality [13].

Wavelet transform, as a useful analytic tool for unstable signals, is also widely applied to artificial intelligence, such as computer classification and recognition, artificial synthesis of music and language, medical imaging and diagnosis and so on. Especially, wavelet transform performs well in boundary processing and multi-scale edge detection. Some researchers are devoted to computer vision field. [14] used UNet method to preprocessing under the guidance of medical knowledge. Then, multi-scale receiving field convolution module was used to extract features of the segmented images with different sizes. [15] proposed a novel click-boosted graph ranking

framework for image retrieval, which consists of two coupled components. [16] used an adopted TextRank to extract key sentences and a template-based method to construct questions from key sentences. Then a multi-feature neural network model was built for ranking to obtain the top questions.

Wavelet transform is not only used in signal and image processing, speech recognition and synthesis, but also in digital image encryption algorithms. In the field of digital copyright protection, how to encrypt and protect digital information often involves the encryption and concealment of digital images and their watermark information, as well as the encryption and concealment of video and audio and their watermark information. In terms of the confirmation of digital rights, ledgers and whole-process records are open in blockchain. Digital copyright works are distributedly stored in network nodes by providing decentralized distributed technology. Now blockchain plays a significant role in digital copyright protection, but there are also some limitations of blockchain in the aspects of privacy protection, data processing and data storage. There are many researchers devoted to the study of privacy protection and data security based on cloud computing and blockchain. The enhanced secure access schemes for outsourced and a secure deduplication scheme proposed in [17-20] can effectively protect local data. [21] proposed one secure data integrity verification scheme for cloud storage. [22-24] proposed a secure blockchain-based schemes for IoT data credibility, a secure data storage and recovery in industrial blockchain network environments and a blockchain-based scheme to protect data confidentiality and traceability.

Except for the case of order one, B-splines of orders greater than one are not orthogonal. Methods that can make wavelets obtain orthogonality must be applied to construct compactly supported orthogonal wavelet and scaling filters. However, when constructing the compactly supported orthogonal wavelets for general purposes, applying other methods to get orthogonality makes construction complicated. Therefore, we construct the compactly supported orthogonal wavelet filters by solving nonlinear equations about filter coefficients with optimization methods and obtaining a series of wavelet filters with different linear phases and multiresolution properties.

We construct compactly supported orthogonal wavelet filters based on an elementary method in [25]. Bi-scale equation and the similarity between orthogonality condition and the formula $\left(cos^2\frac{\omega}{2}+sin^2\frac{\omega}{2}\right)^2=1$ are used to construct a system of nonlinear equations about filter coefficients of compactly supported orthogonal wavelets. We derive a system of nonlinear equations about filter coefficients using double-scale equation and properties of wavelet and scaling filters. Most of the existing methods for solving the system of nonlinear equations are Newton Iteration Method and its improved forms. However, Newton Iteration Method has a fatal shortcoming -- dependence of the solution on the initial values. That is, if the initial values are not correctly selected, likely, the solution of the system of nonlinear equations will not be obtained. The least-square method can expand the selection range of initial values, and sometimes for the same initial values, the Newton iteration method may not converge to the correct solution, but the least square method can [26]. Based on the Gauss-Newton method for the least-square problem, we propose optimization algorithms to solve the system of nonlinear equations. When the given initial values of filter coefficients vary, solutions to the system of nonlinear equations are different. Thus, we can obtain a series of compactly supported orthogonal wavelets with various features.

Our contributions in constructing compactly supported orthogonal wavelet filters are:

1. deriving system of nonlinear equations about filter coefficients of compactly supported orthogonal wavelets. We derive a system of nonlinear equations about filter coefficients in the case of L=1, L=2 and L=3, then we can obtain compactly supported orthogonal wavelets filters with the length of 4, 6 and 8;
2. proposing optimization algorithms to solve a system of nonlinear equations derived. Based on the Gauss-Newton type method's traditional algorithms, we propose optimization algorithms: 1) Basic Gauss-Newton method; 2) Damping Gauss-Newton method under Wolfe criterion; 3) Gauss-Newton method with QR decomposition; 4) LMF-Dogleg method that adds Dogleg method for trust-region subproblem into LMF method.

In terms of accuracy, iteration times, and complexity of algorithms, we analyze approximate solution results of equations to draw conclusions. We analyze solutions obtained by algorithms of Basic GS method, Damping GS method, GS method with QR decomposition, LMF method, and LMF-Dogleg method to obtain the conclusion. We conclude that the compactly supported orthogonal wavelet bases obtained by the basic Gauss-Newton method are optimal.

The remaining of this article is structured as follows. Section 2 introduces the related concepts and properties of wavelet, the elementary method for constructing compactly supported wavelet filters, and related algorithms for solving the least-square problem. Systems of nonlinear equations about filters coefficients are derived in section3. Section 4 proposes optimization algorithms based on the Gauss-Newton method, and approximate solution results and analysis are presented in section5, and finally, concluding remarks and future directions are given in Section 6.

2. Preliminaries

We present: 1) related concepts and properties of wavelets; 2) an elementary method in [25] is used to construct compactly supported orthogonal wavelet filters in section3; 3) traditional algorithms of Gauss-Newton type method for the least-square problem are used to propose optimization algorithms for solving a system of nonlinear equations in section4.

2.1. Related Concepts and Properties

The following concepts, properties, and theorems are used to construct compactly supported orthogonal wavelet filters in section3.

Definition 1. (Two-scale Equation)

$$\varphi(t) = \sum_n h_n \varphi_{1,n}(t) = \sqrt{2} \sum_n h_n \, \varphi(2t - n) \tag{1}$$

$$\psi(t) = \sum_n g_n \varphi_{1,n}(t) = \sqrt{2} \sum_n g_n \, \varphi(2t - n) \tag{2}$$

Equations (1) and (2) are named two-scale equation $\varphi(t)$ and $\psi(t)$ are standard orthogonal basis functions in scale space V_0 and wavelet space W_0 respectively, and the expansion coefficients are

$$h_n = < \varphi(t), \varphi_{1,n}(t) >$$

$$g_n = < \psi(t), \varphi_{1,n}(t) >$$

The two-scale equation describes the intrinsic and essential relationship between the basis functions of two adjacent scale spaces V_{j-1} and V_j, or adjacent space V and wavelet space W.

Definition 2. (Low-pass Filter)

$$H(\omega) = \frac{1}{\sqrt{2}} \sum_n h_n e^{-in\omega} \tag{3}$$

$\{h_n\}$ is the corresponding coefficient of low-pass filter.

Definition 3. (High-pass Filter)

$$\mathbf{G(\omega) = \frac{1}{\sqrt{2}} \sum_n g_n e^{-in\omega}} \tag{4}$$

$\{g_n\}$ is the corresponding coefficient of high-pass filter.

Theorem 1. $\boldsymbol{H(\omega)}$ and $\boldsymbol{G(\omega)}$ are $\boldsymbol{2\pi}$-periodic functions, and satisfy

$$|H(\omega)|^2 + |H(\omega + \pi)|^2 = 1 \tag{5}$$

$$|G(\omega)|^2 + |G(\omega + \pi)|^2 = 1 \tag{6}$$

Theorem 2.

1)

$$\sum_n h_n = \sqrt{2}, \sum_n g_n = 0 \tag{7}$$

2)

$$\sum_n (-1)^n h_n = 0 \tag{8}$$

3)

$$\sum_{k\in Z} h_{2k} = \sum_{k\in Z} h_{2k+1} = \frac{\sqrt{2}}{2} \tag{9}$$

4)

$$\sum_{k\in Z} |h_k|^2 = 1 \tag{10}$$

2.2. An Elementary Method for Constructing Compactly Supported Orthogonal Wavelets

We construct compactly supported orthogonal wavelets based on an elementary proposed in [25]. Bi-scale equation and the similarity between orthogonality condition and the formula $\left(cos^2\frac{\omega}{2} + sin^2\frac{\omega}{2}\right)^2 = 1$ were used to construct a system of nonlinear equations about filter coefficients of compactly supported orthogonal wavelets. The process of construction is as follows:

Let

$$H(\omega) = \frac{1}{\sqrt{2}}\sum_{n=0}^{M} h_n e^{-in\omega},$$

$$H(0) = 1, |H(\omega)|^2 + |H(\omega+\pi)|^2 = 1,$$

and for the properties of $H(\omega)$, there is

$$|H(\omega)|^2 = \frac{1}{2}\sum_{k=0}^{M} h_k^2 + \sum_{n=1}^{M}(\sum_{j=0}^{M-n} h_j h_{j+n})\, cos\, n\omega = \sum_{n=0}^{M} \zeta_n\, cos\, n\omega,$$

namely

$$|H(\omega)|^2 = \sum_{n=0}^{M} \zeta_n\, cos\, n\omega$$
$$\zeta_0 = \frac{1}{2}\sum_{k=0}^{M} h_k^2\,, \zeta_n = \sum_{j=0}^{M-n} h_j h_{j+n}\,, n = 1,2,\dots,M. \tag{11}$$

To calculate $\{h_n\}$ using formula 11, values of $|H(\omega)|^2$ or $\{\zeta_n\}$ must be given first. In [25], $\{h_n\}$ is calculated by giving the values of $\{\zeta_n\}$, and values of $\{\zeta_n\}$ are derived by using similarity between $\left(cos^2\frac{\omega}{2} + sin^2\frac{\omega}{2}\right)^2 = 1$ and the orthogonality condition $|H(\omega)|^2 + |H(\omega+\pi)|^2 = 1$. Generally, let

$$|H(\omega)|^2 + |H(\omega+\pi)|^2 = 1 = \left(cos^2\frac{\omega}{2} + sin^2\frac{\omega}{2}\right)^{2L}.$$

When L is the general case,

$$|H(\omega)|^2 + |H(\omega+\pi)|^2 = 1 = \left(cos^2\frac{\omega}{2} + sin^2\frac{\omega}{2}\right)^{2L}$$

$$= \sum_{n=0}^{2L}\binom{2L}{n}(cos^2\frac{\omega}{2})^{2L-n}(sin^2\frac{\omega}{2})^n$$

$$= \left(cos^2\frac{\omega}{2}\right)^{2L} + \sum_{n=1}^{L-1}\binom{2L}{n}\left(cos^2\frac{\omega}{2}\right)^{2L-n}\left(sin^2\frac{\omega}{2}\right)^n + \binom{2L}{L}(cos^2\frac{\omega}{2})^L(sin^2\frac{\omega}{2})^L + \sum_{n=L+1}^{2L-1}\binom{2L}{n}\left(cos^2\frac{\omega}{2}\right)^{2L-n}\left(sin^2\frac{\omega}{2}\right)^n + \left(sin^2\frac{\omega}{2}\right)^{2L}$$

$$= (cos^2\frac{\omega}{2})^{2L} + \sum_{n=1}^{L-1}\binom{2L}{n}\left(cos^2\frac{\omega}{2}\right)^{2L-n}\left(sin^2\frac{\omega}{2}\right)^n + \binom{2L}{L}(cos^2\frac{\omega}{2})^{L+1}(sin^2\frac{\omega}{2})^L + \binom{2L}{L}(cos^2\frac{\omega}{2})^L(sin^2\frac{\omega}{2})^{L+1} + \sum_{n=L+1}^{2L-1}\binom{2L}{n}\left(cos^2\frac{\omega}{2}\right)^{2L-n}\left(sin^2\frac{\omega}{2}\right)^n + \left(sin^2\frac{\omega}{2}\right)^{2L}.$$

Let

$$|H(\omega)|^2 = (cos^2\frac{\omega}{2})^{2L} + \sum_{n=1}^{L-1}\binom{2L}{n}\left(cos^2\frac{\omega}{2}\right)^{2L-n}\left(sin^2\frac{\omega}{2}\right)^n + \binom{2L}{L}(cos^2\frac{\omega}{2})^{L+1}(sin^2\frac{\omega}{2})^L, \tag{12}$$

$$|H(\omega+\pi)|^2 = \left(sin^2\frac{\omega}{2}\right)^{2L} + \sum_{n=L+1}^{2L-1}\binom{2L}{n}\left(cos^2\frac{\omega}{2}\right)^{2L-n}\left(sin^2\frac{\omega}{2}\right)^n + \binom{2L}{L}(cos^2\frac{\omega}{2})^L(sin^2\frac{\omega}{2})^{L+1}. \tag{13}$$

2.3. The Least-Squares Problem (LS)[27]

Definition 4. Giving a set of experimental data $(\boldsymbol{t_i}, \boldsymbol{y_i})(\boldsymbol{i} = \boldsymbol{1}, \dots, \boldsymbol{m})$ and a functional model $\boldsymbol{f}(\boldsymbol{x}; \boldsymbol{t})$, remaining quantity $\boldsymbol{r_i}(\boldsymbol{x})$ is $\boldsymbol{r_i}(\boldsymbol{x}) = \boldsymbol{y_i} - \boldsymbol{f}(\boldsymbol{x}; \boldsymbol{t_i}), \quad \boldsymbol{i} = \boldsymbol{1}, \dots, \boldsymbol{m}$. The least-squares problem is

$$\min f(x) = \frac{1}{2}\sum_{i=1}^{m} r_i^2(x) = \frac{1}{2} r(x)^T r(x), \quad x \in \mathbb{R}^n, \quad m \geq n. \tag{14}$$

Definition 5. The Newton equation for solving the least-squares problem is

$$(J_k^T J_k + S_k) d_k = -J_k^T r_k, \tag{15}$$

where

$$J(x) = [\nabla r_1(x), \dots, \nabla r_m(x)]^T \in \mathbb{R}^{m \times n},$$

$$S(x) = \sum_{i=1}^{m} r_i(x) \nabla^2 r_i(x), \tag{16}$$

$$J_k = J(x_k), S_k = S(x_k).$$

The methods of solving the least-squares problem can also be used to get approximate solution $x = [x_1, x_2, \dots, x_n]^T$ of a system of nonlinear equations

$$r(x) = [r_1(x), \dots, r_m(x)]^T = 0.$$

2.4. Gauss-Newton Method

Definition 6. Gauss-Newton Equation

$$J_k^T J_k d_k = -J_k^T r_k. \tag{17}$$

The advantage of the Gauss-Newton method to solve the system of nonlinear equations lies is that it does not need to calculate the second derivative of $r(x)$[27]. So the minimum point d_k in the linear least-squares problem with respect to d can be obtained by computing the value of $d_k = -(J_k^T J_k)^{-1} J_k^T r_k$.

The following is the traditional algorithm of the Gauss-Newton method:

Algorithm 1. Gauss-Newton Method to Solve LS[27]

Step 1. Give $x_0, \varepsilon > 0, k = 0$;
Step 2. if the termination condition is satisfied, stop the iteration;
Step 3. solve $J_k^T J_k d = -J_k^T r_k$ to obtain d_k ;
Step 4. compute $x_{k+1} = x_k + \alpha_k d_k$, $k = k + 1$, go back to Step 2.

2.5. QR Decomposition of $\mathbf{J_k}$

To reduce the solution sensitivity caused by the rounding error in solving LS with the basic Gauss-Newton method, and improve the feasibility of the solution process and accuracy of the final solution, in [27], QR decomposition of J_k were used to obtain d_k. Solving the Gauss-Newton equation is equivalent to solve

$$min\frac{1}{2}\parallel Q_k^T J_k d + Q_k^T r_k \parallel^2.$$

First, QR decomposition is used to obtain

$$J_k = Q_k \begin{bmatrix} R_k \\ 0 \end{bmatrix}, \quad (18)$$

where $Q_k \in \mathbb{R}^{m\times m}$ is an orthogonal matrix, $R_k \in \mathbb{R}^{n\times n}$ is the upper triangular matrix with non-zero diagonal elements and $0 \in \mathbb{R}^{(m-n)\times n}$ is a zero matrix.

Then partition Q_k to get

$$Q_k = \begin{bmatrix} Q_1^{(k)} & Q_2^{(k)} \end{bmatrix}, \quad (19)$$

where $Q_1^{(k)} \in \mathbb{R}^{m\times n}$, $Q_2^{(k)} \in \mathbb{R}^{m\times(m-n)}$.

And let

$$Q_k^T r_k = \begin{bmatrix} Q_1^{(k)^T} \\ Q_2^{(k)^T} \end{bmatrix} r_k = \begin{bmatrix} b1 \\ b2 \end{bmatrix}_{m-n}^{n}, \quad (20)$$

so

$$\begin{aligned} \parallel J_k d + r_k \parallel^2 &= \parallel Q_k^T J_k d + Q_k^T r_k \parallel^2 \\ &= \parallel \begin{bmatrix} R_k \\ 0 \end{bmatrix} d + Q_k^T r_k \parallel^2 \quad (21) \\ &= \parallel R_k d + b_1 \parallel^2 + \parallel b_2 \parallel^2. \end{aligned}$$

d_k is the solution of the least-squares problem if and only if d_k is the solution of $R_k d = -b_1$.

2.6. LMF Method

Definition 7. LMF Equation

$$(J_k^T J_k + v_k I)d = -J_k^T r_k \quad (22)$$

To solve the situation that $J_k^T J_k$ is singular in the iteration process with the Gauss-Newton method, Levenberg proposed the LMF equation to obtain d_k, where $v_k \geq 0$. LM method is a trust region method, and the value of v_k can be modified in iterations with the idea of the trust region method. LMF method solves the least-squares problem with the following algorithm:

Algorithm 2. LMF Method to Solve LS [16]

Step 1. Give $x_0 \in \mathbb{R}^n, v_0 > 0, \varepsilon > 0, k = 0$;
Step 2. if the termination condition $\| g(x_k) \| \leq \varepsilon$ is satisfied, stop the iteration;
Step 3. solve $(J_k^T J_k + v_k I)d = -J_k^T r_k$ to obtain d_k;
Step 4. compute r_k with $r_k = \frac{\Delta f_k}{\Delta q_k}$;
Step 5. if $r_k < 0.25$, $v_{k+1} = 4v_k$; else if $r_k > 0.75$, $v_{k+1} = \frac{v_k}{2}$; else $v_{k+1} = v_k$;
Step 6. if $r_k \leq 0$, $x_{k+1} = x_k$; else $x_{k+1} = x_k + d_k$, $k = k + 1$ go back to Step 2.

And

$$\Delta f_k = f(x_k) - f(x_k + d_k),$$

$$\Delta q_k = q_k(0) - q_k(d_k) = \frac{1}{2} d_k^T (v_k d_k - g_k), \tag{23}$$

$$g_k = J_k^T r_k.$$

2.7. Dogleg Method

Powell proposed the Dogleg method to solve the trust-region subproblem in the LMF method, since the direction d_k^{LM} obtained from the LM equation is affected by the value of v_k, which makes solution speedy varied greatly. The algorithm of the Dogleg method is:

Algorithm 3. Dogleg Method to Solve Trust Region Subproblem

Step 1. Give $\Delta_k > 0, J_k, r_k$;
Step 2. if $\| d_k^{GN} \| \leq \Delta_k$, $d_k = d_k^{GN}$, and output d_k, stop the iteration;
Step 3. compute $\alpha_k = \frac{\| d_k^{SD} \|^2}{\| J_k d_k^{SD} \|^2}$; if $\alpha_k \| d_k^{SD} \| \geq \Delta_k$,
$d_k = \frac{\Delta_k}{\| d_k^{SD} \|} d_k^{SD}$, and output d_k;
Step 4. solve the unary quadratic equation
$\| d_k^{GN} - \alpha_k d_k^{SD} \|^2 \beta^2 + 2\alpha_k {d_k^{SD}}^T (d_k^{GN} - \alpha_k d_k^{SD})\beta + \alpha_k^2 \| d_k^{SD} \|^2 - \Delta_k^2$ to obtain β (take the solution greater than 0) ;
Step 5. compute $d_k = (1 - \beta)\alpha_k d_k^{SD} + \beta d_k^{GN}$, and output d_k, stop the iteration.

And d_k^{GN} is Gauss-Newton direction, i.e.

$$d_k^{GN} = -(J_k^T J_k)^{-1} J_k^T r_k, \quad d_k^{SD} = -J_k^T r_k. \tag{24}$$

3. Construction of a system of nonlinear equations

Based on the method presented in subsection 2.2, we derive a system of nonlinear equations about filter coefficients of compactly supported wavelets in the case of L=1, L=2 and L=3 to obtain filters with the length of 4, 6, and 8. In subsection 2.2, $|H(\omega)|^2$ can be presented as

$$|H(\omega)|^2 = (cos^2\frac{\omega}{2})^{2L} + \sum_{n=1}^{L-1}\binom{2L}{n}\left(cos^2\frac{\omega}{2}\right)^{2L-n}\left(sin^2\frac{\omega}{2}\right)^{n} + \binom{2L}{L}(cos^2\frac{\omega}{2})^{L+1}(sin^2\frac{\omega}{2})^{L}.$$

When the value of L is given, $|H(\omega)|^2$ can be presented as

$$|H(\omega)|^2 = \sum_{n=0}^{M}\zeta_n \cos n\omega$$

to obtain values of $\{\zeta_n\}$, which are used to construct a system of nonlinear equations about $\{h_n\}$. Derivations of values of $\{\zeta_n\}$ in the case of L=1, L=2, and L=3 are as follows:

3.1. L=1

When L=1, let

$$\begin{aligned}|H(\omega)|^2 &= cos^4\frac{\omega}{2} + 2cos^4\frac{\omega}{2}sin^2\frac{\omega}{2} = \left(\frac{1+cos\omega}{2}\right)^2 + \frac{1}{4}sin^2\omega(1+cos\omega)\\ &= \frac{1}{4}(1+2cos\omega+cos^2\omega+sin^2\omega+cos\omega sin^2\omega)\\ &= \frac{1}{4}\left[2+2cos\omega+\frac{1}{4}(cos\omega sin^2\omega+cos\omega sin^2\omega)\right]\\ &= \frac{1}{2}+\frac{1}{2}cos\omega+\frac{1}{16}[cos\omega(1-cos2\omega)+sin\omega sin2\omega]\\ &= \frac{1}{2}+\frac{1}{2}cos\omega+\frac{1}{16}(cos\omega-cos\omega cos2\omega+sin\omega sin2\omega)\\ &= \frac{1}{2}+\frac{9}{16}cos\omega-\frac{1}{16}cos3\omega\end{aligned}$$

then we have values of $\{\zeta_n\}$ in the case of L=1,

$$\{\zeta_0,\zeta_1,\zeta_2,\zeta_3\} = \left\{\frac{1}{2},\frac{9}{16},0,-\frac{1}{16}\right\}. \quad \textbf{(25)}$$

3.2. L=2

When L=2, let

$$|H(\omega)|^2 = cos^8\frac{\omega}{2} + 4cos^6\frac{\omega}{2}sin^2\frac{\omega}{2} + 6cos^6\frac{\omega}{2}sin^4\frac{\omega}{2}$$

$$= \left(\frac{1+cos\omega}{2}\right)^4 + 4\left(\frac{1+cos\omega}{2}\right)^2 sin^2\frac{\omega}{2}cos^2\frac{\omega}{2} + 6cos^2\frac{\omega}{2}cos^4\frac{\omega}{2}sin^4\frac{\omega}{2}$$

$$= \frac{1}{16}[(1+cos\omega)^4 + 4(1+cos\omega)^2sin^2\omega + 3(1+cos\omega)sin^4\omega]$$

$$= \frac{1}{16}(cos^4\omega + 4cos^3\omega + 6cos^2\omega + 4cos\omega + 1 + 4cos^2\omega sin^2\omega + 8cos\omega sin^2\omega + 4sin^2\omega + 3cos\omega sin^4\omega + 3sin^4\omega)$$

$$= \frac{1}{16}(1 + 4cos\omega + cos^4\omega + cos^2\omega sin^2\omega + 3cos^2\omega sin^2\omega + 3sin^4\omega + 4cos^3\omega + 6cos^2\omega + 4sin^2\omega + 8cos\omega sin^2\omega + 3cos\omega sin^4\omega)$$

$$= \frac{1}{16}(1 + 4cos\omega + 7cos^2\omega + 7sin^2\omega + 4cos\omega + 4cos\omega sin^2\omega + 3cos\omega sin^4\omega)$$

$$= \frac{1}{16}(8 + 8cos\omega + 4cos\omega sin^2\omega + 3cos\omega sin^4\omega)$$

$$= \frac{1}{2} + \frac{1}{16}\left(8cos\omega - \frac{25}{16}cos3\omega + \frac{3}{16}cos5\omega + \frac{22}{16}cos\omega\right)$$

$$= \frac{1}{2} + \frac{75}{128}cos\omega - \frac{25}{256}cos3\omega + \frac{3}{256}cos5\omega,$$

then we have values of $\{\zeta_n\}$ in the case of L=2,

$$\{\zeta_0, \zeta_1, \zeta_2, \zeta_3, \zeta_4, \zeta_5\} = \left\{\frac{1}{2}, \frac{75}{128}, 0, -\frac{25}{256}, 0, \frac{3}{256}\right\}. \quad \textbf{(26)}$$

3.3. L=3

When L=3, let

$$|H(\omega)|^2 = (cos^2\frac{\omega}{2})^6 + \sum_{n=1}^{2}\binom{6}{n}(cos^2\frac{\omega}{2})^{6-n}(sin^2\frac{\omega}{2})^n + \binom{6}{3}(cos^2\frac{\omega}{2})^4(sin^2\frac{\omega}{2})^3$$

$$= cos^{12}\frac{\omega}{2} + 6cos^{10}\frac{\omega}{2}sin^2\frac{\omega}{2} + 15cos^8\frac{\omega}{2}sin^4\frac{\omega}{2} + 20cos^8\frac{\omega}{2}sin^6\frac{\omega}{2}$$

$$= (\frac{1+cos\omega}{2})^6 + 6(\frac{1+cos\omega}{2})^4 cos^2\frac{\omega}{2}sin^2\frac{\omega}{2} + 15(\frac{1+cos\omega}{2})^2 cos^4\frac{\omega}{2}sin^4\frac{\omega}{2} + 20(\frac{1+\cos\omega}{2})cos^6\frac{\omega}{2}sin^6\frac{\omega}{2}$$

$$= \frac{1}{64}[(1+cos\omega)^6 + 6(1+cos\omega)^4 sin^2\omega + 15(1+cos\omega)^2 sin^4\omega + 10(1 + cos\omega)sin^6\omega]$$

$$= \frac{1}{64}\left[\sum_{k=0}^{6}\binom{6}{k}(cos\omega)^{6-k} + 6sin^2\omega\sum_{k=0}^{4}\binom{4}{k}(cos\omega)^{4-k} + 15(1+2cos\omega+cos^2\omega)sin^4\omega + 10cos\omega sin^6\omega + 10sin^6\omega\right]$$

$$= \frac{1}{64}(1 + 6cos\omega + 15cos^2\omega + 6sin^2\omega + 20cos^3\omega + 24cos\omega sin^2\omega + 15cos^4\omega + 15sin^4\omega + 36cos^2\omega sin^2\omega + 6cos^5\omega + 30cos\omega sin^4\omega + 24cos^3\omega sin^2\omega + cos^6\omega + 10sin^6\omega + 6cos^4\omega sin^2\omega + 15cos^2\omega sin^4\omega + 10cos\omega sin^6\omega)$$

$$= \frac{1}{64}(7 + 26cos\omega + 24cos^2\omega + 15sin^2\omega + 6cos^3\omega + 22cos\omega sin^2\omega + cos^4\omega + 10sin^4\omega + 11cos^2\omega sin^2\omega + 12cos\omega sin^4\omega + 10cos\omega sin^6\omega)$$

$$= \frac{1}{64}(22 + 32cos\omega + 10cos^2\omega + 10sin^2\omega + 16cos\omega sin^2\omega + 12cos\omega sin^4\omega + 10cos\omega sin^6\omega)$$

$$= \frac{1}{64}(32 + 32cos\omega + 16cos\omega sin^2\omega + 12cos\omega sin^4\omega + 10cos\omega sin^6\omega)$$

$$= \frac{1}{2} + \frac{1}{64}\left(\frac{1225}{32}cos\omega - \frac{245}{32}cos3\omega + \frac{49}{32}cos5\omega - \frac{5}{32}cos7\omega\right)$$

$$= \frac{1}{2} + \frac{1225}{2048}cos\omega - \frac{245}{2048}cos3\omega + \frac{49}{2048}cos5\omega - \frac{5}{2048}cos7\omega,$$

then we have values of $\{\zeta_n\}$ in the case of L=3,

$$\{\zeta_0,\zeta_1,\zeta_2,\zeta_3,\zeta_4,\zeta_5,\zeta_6,\zeta_7\}=\left\{\frac{1}{2},\frac{1225}{2048},0,-\frac{245}{2048},0,\frac{49}{2048},0,-\frac{5}{2048}\right\}. \tag{27}$$

Then use formula (11) and theorem 2.2 to conduct a system of nonlinear equations about filter coefficients $\{h_n\}$ in the case of L=1, L=2 and L=3 as follows:

when L=1,

$$\begin{cases} h_0^2+h_1^2+h_2^2+h_3^2=1 \\ h_0h_1+h_1h_2+h_2h_3=\dfrac{9}{16} \\ h_0h_2+h_1h_3=0 \\ h_0h_3=-\dfrac{1}{16} \\ h_0+h_1+h_2+h_3=\sqrt{2} \end{cases}$$

when L=2,

$$\begin{cases} h_0^2+h_1^2+h_2^2+h_3^2+h_4^2+h_5^2=1 \\ h_0h_1+h_1h_2+h_2h_3+h_3h_4+h_4h_5=\dfrac{75}{128} \\ h_0h_2+h_1h_3+h_2h_4+h_3h_5=0 \\ h_0h_3+h_1h_4+h_2h_5=-\dfrac{25}{256} \\ h_0h_4+h_1h_5=0 \\ h_0h_5=\dfrac{3}{256} \\ h_0+h_1+h_2+h_3+h_4+h_5=\sqrt{2} \end{cases}$$

when L=3,

$$\begin{cases} h_0^2+h_1^2+h_2^2+h_3^2+h_4^2+h_5^2+h_6^2+h_7^2=1 \\ h_0h_1+h_1h_2+h_2h_3+h_3h_4+h_4h_5+h_5h_6+h_6h_7=\dfrac{1225}{2048} \\ h_0h_2+h_1h_3+h_2h_4+h_3h_5+h_5h_7=0 \\ h_0h_3+h_1h_4+h_2h_5+h_3h_6+h_4h_7=-\dfrac{245}{2048} \\ h_0h_4+h_1h_5+h_2h_6+h_3h_7=0 \\ h_0h_5+h_1h_6+h_2h_7=\dfrac{49}{2048} \\ h_0h_6+h_1h_7=0 \\ h_0h_7=-\dfrac{5}{2048} \\ h_0+h_1+h_2+h_3+h_4+h_5+h_6+h_7=\sqrt{2} \end{cases}$$

Compactly supported orthogonal wavelet filters with the length of 4, 6, and 8 can be obtained by approximately solving the abovementioned three systems of nonlinear

equations with filter coefficients $\{h_n\}$. In the following section, we propose optimization algorithms for the approximate solution of the system of nonlinear equations.

4. Optimization Algorithms

To obtain optimal compactly supported orthogonal wavelet bases by solving the system of nonlinear equations constructed in section3, we propose optimization algorithms of variants of the Gauss-Newton method. When the dimension of the system of nonlinear equations is small, the Gauss-Newton type method can locally converge to the solution of a system of nonlinear equations fast. Since the upper dimension of the nonlinear system constructed in section 3 is only 8, we propose the Basic Gauss-Newton method, Damping Gauss-Newton method and QR decomposition Method based on GN method, and Dogleg method based on LMF method.

4.1. Basic Gauss-Newton Method

The basic Gauss-Newton method refers to the Gauss-Newton method on $\alpha_k = 1$, as its most significant advantage, is that the algorithm is simple and easy to be implemented. The basic Gauss-Newton method solves the least-squares problem with the following algorithm:

Algorithm 4. Basic Gauss-Newton Method to Solve LS

Step 1. Give $x_0, \varepsilon > 0, k = 0$;
Step 2. if the termination condition $\| g(x_k) \| \leq \varepsilon$ is satisfied, stop the iteration;
Step 3. solve $J_k^T J_k d = -J_k^T r_k$ to obtain d_k ;
Step 4. compute $x_{k+1} = x_k + d_k, \quad k = k + 1,$ go back to Step 2.

4.2. Damping Gauss-Newton Method.

The damping Gauss-Newton method refers to the Gauss-Newton method with line search. We use the inexact line search under the Wolfe criterion to obtain α_k, namely let $\beta \epsilon (0,1),\ \sigma \epsilon (0,0.5),\ \alpha_k = \beta^{m_k}$, where $m_k$is the minimal nonnegative integer satisfying the following inequality:

$$f(x_k + \beta^m d_k) \leq f(x_k) + \sigma \beta^m g_k^T d_k. \qquad (28)$$

Compared with the basic Gauss-Newton method, the damping Gauss-Newton method has higher precision due to line search to find the step size and is relatively more complex. The algorithm of the damping Gauss-Newton method to solve LS is as follows:

Algorithm 5. Damping Gauss-Newton Method to Solve LS

Step 1. Give $x_0, \varepsilon > 0, k = 0$;
Step 2. if the termination condition $\| g(x_k) \| \leq \varepsilon$ is satisfied, stop the iteration;
Step 3. solve $J_k^T J_k d = -J_k^T r_k$ to obtain d_k;
Step 4. give $\beta \epsilon (0,1)$, $\sigma \epsilon (0,0.5)$, m_k=0;
Step 5. if the current m_k satisfies $f(x_k + \beta^m d_k) \leq f(x_k) + \sigma \beta^m g_k^T d_k$, stop the iteration;
Step 6. or else $m = m + 1$, go back to Step5;
Step 7. compute $\alpha_k = \beta^{m_k}$;
Step 8. compute $x_{k+1} = x_k + \alpha_k d_k$, $k = k + 1$, go back to Step 2.

4.3. Gauss-Newton Method with QR Decomposition.

Based on QR decomposition of J_k, d_k can be obtained by solving the equation $R_k d = -b_1$, which reduces the solution sensitivity caused by the rounding error in solving LS with the Gauss-Newton method, and improves the feasibility of the solution process and the accuracy of the final solution. The QR decomposition iteration method is used to solve the least-square problem with the following algorithm:

Algorithm 6. Gauss-Newton Method with QR Decomposition to Solve LS

Step 1. Give $x_0, \varepsilon > 0, k = 0$;
Step 2. if the termination condition $\| g(x_k) \| \leq \varepsilon$ is satisfied, stop the iteration;
Step 3. compute the QR decomposition of J_k;
Step 4. compute $b_1 = {Q_1^{(k)}}^T r_k$;
Step 5. solve the upper triangular equation $R_k d = -b_1$ to obtain d_k;
Step 6. compute $x_{k+1} = x_k + d_k$, $k = k + 1$, go back to Step 2.

4.4. LMF-Dogleg Method

Based on the LMF method, we propose the LMF-Dogleg method that uses Dogleg method to solve the subproblem of selecting d_k in the LMF method. LMF-Dogleg method is used to solve the least-square problem with the following algorithm:

Algorithm 7. LMF-Dogleg Method to Solve LS

Step 1. Give $x_0 \in \mathbb{R}^n, \Delta_k > 0, \varepsilon > 0, k = 0$;
Step 2. if the termination condition $\| g(x_k) \| \leq \varepsilon$ is satisfied, stop the iteration;
Step 3. if $\| d_k^{GN} \| \leq \Delta_k$, $d_k = d_k^{GN}$, and output d_k;
Step 4. compute $\alpha_k = \frac{\| d_k^{SD} \|^2}{\| J_k d_k^{SD} \|^2}$; if $\alpha_k \| d_k^{SD} \| \geq \Delta_k$ and $d_k = \frac{\Delta_k}{\| d_k^{SD} \|} d_k^{SD}$,output d_k;
Step 5. solve the unary quadratic equation

$\| d_k^{GN} - \alpha_k d_k^{SD} \|^2 \beta^2 + 2\alpha_k {d_k^{SD}}^T (d_k^{GN} - \alpha_k d_k^{SD})\beta + \alpha_k^2 \| d_k^{SD} \|^2 - \Delta_k^2$ to obtain β (take the solution greater than 0);

Step 6. compute $d_k = (1-\beta)\alpha_k d_k^{SD} + \beta d_k^{GN}$, and output d_k;

Step 7. update Δ_k: compute $r_k = \frac{\Delta f_k}{\Delta q_k}$; if $r_k < 0.25$, $\Delta_{k+1} = \Delta_k/4$; else if $r_k > 0.75$, $\Delta_{k+1} = 2\Delta_k$; $\Delta_{k+1} = \Delta_k$;

Step 8. if $r_k \leq 0$, $x_{k+1} = x_k$; else $x_{k+1} = x_k + d_k$, and $k = k+1$, go back to step 2.

5. Analysis

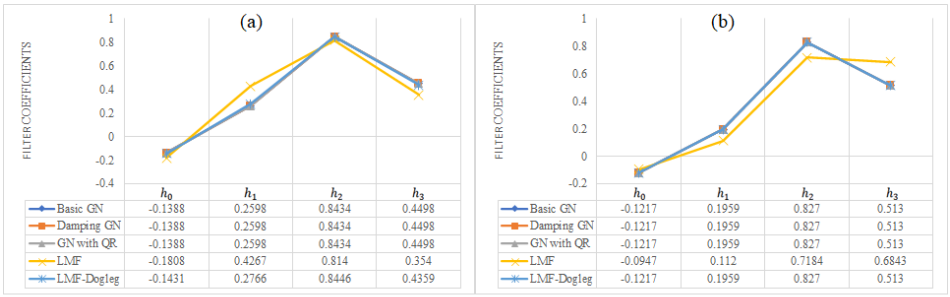
To obtain the optimal compactly supported orthogonal wavelet bases with lengths of 4, 6, and 8, we implement the optimization algorithms proposed in section 4 by MATLAB to solve the system of nonlinear equations derived in section3. Giving two sets of initial values of $\{h_n\}$, we obtain different solutions of the three systems of nonlinear equations. Then we analyze the approximate solution results to obtain the best compactly supported orthogonal wavelet bases constructed based on the elementary method proposed in [8]. We remain to solve precision $\varepsilon = 0.001$, trust region radius $\Delta_0 = 2$, and $v_0 = 10$ unchanged, and only make the initial values of $\{h_n\}$ vary. The initial values and solutions are presented as follows:

1. When L=1, let the initial values $h_1^{(0)} = \{0,1,1,1\}$ and $\{0,0,1,1\}$ respectively. The solving results are presented in Fig.1, which gives two sets of graphs corresponding to different initial values and explicitly indicate the solution's structure. Furthermore, the data of two tables in Fig.1 are presented to analyze the accuracy of algorithms.
2. When L=2, let the initial values $h_2^{(0)} = \{1,0,1,1,1,1\}$ and $\{0,0,0,1,1,1\}$ respectively. The solving results are presented in Fig.2, where it is plain that five curves of solution are almost coincident.
3. When L=3, let the initial values $h_3^{(0)} = \{1,0,1,1,1,1,1,1\}$ and $\{0,0,0,0,1,1,1,1\}$ respectively. The solving results are presented in Fig.3. It can be seen from the two graphs that solution curves of different algorithms are almost coincident except for the LMF. The detailed analyses about this phenomenon are discussed as follows.

The selection of initial values and optimization algorithms affects the accuracy and solving speed of approximate solutions, as can be seen in Fig.1, Fig.2, and Fig.3 that the structure of the solution varies with algorithms and initial values. Therefore, we analyze the solution results and figure out under what circumstances the method can be applied to obtain optimal compactly support orthogonal wavelet bases. We analyze the solution results on the accuracy, iteration times (of different cases presented in Table1), solution speed, and algorithms' complexity. The accuracy of the approximate solutions is analyzed based on the degree of coincidence between the solution results and the constant terms in the original equation group (the checking method is to substitute the solution results into the left side of the system of nonlinear equations and compare the values obtained with the right side).

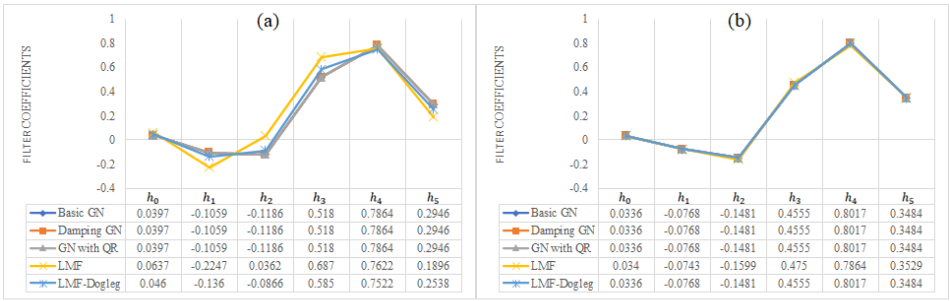
Table 1. Data in the table are iterations of different solving processes in Fig.1, Fig.2, and Fig.3, given by MATLAB programs. They are collected to measure the solving speed and complexity of different algorithms

Initial Values / Algorithms	L=1.(a)	L=1.(b)	L=2.(a)	L=2.(b)	L=3.(a)	L=3.(b)
Basic GN	8	6	11	5	8	8
Damping GN	8	6	11	5	11	8
GN with QR	8	6	11	5	8	8
LMF	9	6	10	7	18	11
LMF-Dogleg	8	6	10	5	7	10



L = 1

Fig.1. To analyze the influence of variations about initial values of filter coefficients on approximate solution results. Among the initial values, solving precision $\varepsilon = 0.001$, trust-region radius $\Delta_0 = 2$, and $v_0 = 10$ remain unchanged, and only the initial filter coefficients are variables. The initial values of filter coefficients of (a) are {0,1,1,1}, and {0,0,1,1} of (b)



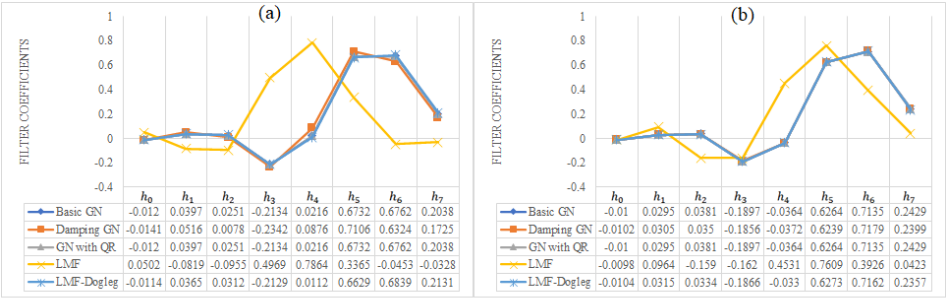
L = 2

Fig.2. Solving precision $\varepsilon = 0.001$, trust region radius $\Delta_0 = 2$, and $v_0 = 10$ remain unchanged. The initial values of filter coefficients of (a) are {1,0,1,1,1,1}, and {0,0,0,1,1,1} of (b). It can be seen in (b) that five curves of solution are almost coincident, and compared with (a), the structure of solution in (a) are basically tallies (b)

Analysis of the solution speed is based on the solution process's iteration times, which is presented in Table 1 (given by the MATLAB program). The iteration times are the number of iterations used until the termination condition $\| g(x_k) \| \leq \varepsilon$ is satisfied.

The solution results of (a) in each figure show that when using various methods to approximate the solution, the difference between the structure obtained and the iteration coefficient is significant; the results of (b) in Fig.1, Fig.2, and Fig.3 relatively show a small or even no difference between most of them. Accordingly, the results in the chart(a) in each figure serve as group1, and results in chart(b) serve as group2 to analyze the accuracy and solve the speed of different algorithms.

The features of initial values selected in the group1 are: 1) there is only one 0 element; 2) the rest elements are all 1. We can see from Fig.1, Fig.2, and Fig.3 that iteration times of LMF are more than other algorithms: 1) when L=1 and 2, the number of iterations of the Gauss-Newton method is less than or equal to the LMF-Dogleg method; 2) when L=3, iterations times of the LMF-Dogleg solution is lower than the Gauss-Newton method. After checking the results in group1, we obtain conclusions: 1) when L=1 and 2, the accuracy of the Gauss-Newton method is better than that of the LMF method and LMF-Dogleg method; 2) when L=3, LMF-Dogleg method, the LMF method, and damping Gauss-Newton method are similar in accuracy and higher than the basic Gauss-Newton method. In addition, the structure of approximate solution results of the LMF method is different from that of other methods.



	h_0	h_1	h_2	h_3	h_4	h_5	h_6	h_7
Basic GN	-0.012	0.0397	0.0251	-0.2134	0.0216	0.6732	0.6762	0.2038
Damping GN	-0.0141	0.0516	0.0078	-0.2342	0.0876	0.7106	0.6324	0.1725
GN with QR	-0.012	0.0397	0.0251	-0.2134	0.0216	0.6732	0.6762	0.2038
LMF	0.0502	-0.0819	-0.0955	0.4969	0.7864	0.3365	-0.0453	-0.0328
LMF-Dogleg	-0.0114	0.0365	0.0312	-0.2129	0.0112	0.6629	0.6839	0.2131

	h_0	h_1	h_2	h_3	h_4	h_5	h_6	h_7
Basic GN	-0.01	0.0295	0.0381	-0.1897	-0.0364	0.6264	0.7135	0.2429
Damping GN	-0.0102	0.0305	0.035	-0.1856	-0.0372	0.6239	0.7179	0.2399
GN with QR	-0.01	0.0295	0.0381	-0.1897	-0.0364	0.6264	0.7135	0.2429
LMF	-0.0098	0.0964	-0.159	-0.162	0.4531	0.7609	0.3926	0.0423
LMF-Dogleg	-0.0104	0.0315	0.0334	-0.1866	-0.033	0.6273	0.7162	0.2357

L = 3

Fig.3. The initial values of filter coefficients of (a) are {1,0,1,1,1,1,1,1} and {0,0,0,0,1,1,1,1} of (b). Solving precision $\varepsilon = 0.001$, trust region radius $\Delta_0 = 2$, and $v_0 = 10$ remain unchanged. It can be seen in the two charts that curves of solutions are almost coincident except for the LMF

The feature of initial values selected in the group2 is that half of the elements of $\{h_n\}$ are 0, and the others are all 1. The solution results show that the accuracy of the group2 is higher than group1. Therefore, the initial values selected in group2 can be regarded as close to the exact solution. We can see from these three tables: 1) When L=1 and 2, only the results of LMF methods are different from others, and the number of iterations is higher or equal to others; 2) When L=3, the relation of iteration times is: LMF > LMF-Dogleg > Gauss-Newton type method. The checking results show that in three cases, the LMF method has the lowest accuracy; 3) When L=3, the LMF-Dogleg method and damping Gauss-Newton method are more accurate than other methods, which are similar to group1.

Synthesizing the above analysis, we can see that although the differences between initial values have effects on the solution results, in most cases, the algorithms in this paper can converge to the solution when approximately solving the system of nonlinear equations. After checking, we find that these effects are not noticeable: the difference of

precision is only 0.001, and the difference of iteration times on L=1 and L=3 is minimal.

In addition, compared with the Newton Iteration method, it is more efficient to solve the system of nonlinear equations about filter coefficients by the least square method, and the solution process is relatively stable and efficient. Moreover, the efficiency of the LMF method is inferior in all aspects, and as with the improved LMF method, the LMF-Dogleg is superior to the LMF method in all aspects of analysis. The Gauss-Newton method was superior to the LMF-Dogleg method when L=1 and L=2, and slightly inferior to the LMF-Dogleg method and Gauss-Newton method when L=3, but the difference between the three methods was very small. In addition, the LMF-Dogleg method and damping Gauss-Newton method are more complicated than the basic Gauss-Newton method, and in the process of the actual execution of the program, the cost of time and space is more than the basic Gauss-Newton method. However, the precision and iterative coefficients are almost identical. We can see the basic Gauss-Newton method is easy to implement and has high precision. Therefore, we can conclude that the wavelet filter bases that are constructed by solving a system of nonlinear equations about filter coefficients with the basic Gauss-Newton method are optimum.

6. Conclusions and Future Work

We construct compactly supported orthogonal wavelet filters with different linear phases and multiresolution properties by solving the systems of nonlinear equations about filter coefficients. We drive these systems of nonlinear equations about filter coefficients of wavelets using bi-scale equation and similarity between orthogonality condition and the formula $\left(cos^2\frac{\omega}{2}+sin^2\frac{\omega}{2}\right)^2=1$. To approximately solve these systems of nonlinear equations, we propose optimization algorithms of the Basic Gauss-Newton method, Damping Gauss-Newton method, Gauss-Newton method with QR decomposition, and LMF-Dogleg method. We then analyze the solution results by these algorithms on the accuracy, iteration times, solution speed, and complexity of algorithms and conclude that the wavelet filters are constructed by a solving system of nonlinear equations about filter coefficients with basic Gauss-Newton method are optimal.

The general form of a system of nonlinear equations was not be given in this paper. As the derivation has reached 16 power when L=4, the construction is too complicated, and we cannot guarantee the correctness of the nonlinear system's constant term. Therefore, we only construct compactly supported wavelet filters in the cases of L=1, L=2, and L=3. Nevertheless, we obtain some laws :

1. There are 2L+2 unknown numbers and 2L+3 equations in the system of the nonlinear equation;
2. The general simplified result is: $|H(\omega)|^2=\frac{1}{2}+\sum_{k=0}^{L}(-1)^L\alpha_k\cos(2k+1)\omega$, where α_k is a coefficient.

As future work, we will attempt to find the general laws between the filter coefficients, and construct a system of nonlinear equations about them, then design a general program that can output the filter coefficients only by giving the value of L. In

this way, we can improve the efficiency of constructing compactly supported orthogonal wavelet filters with various features, which are widely applied to signal analysis and processing in multimedia and other fields.

Acknowledgment. This work was supported by Beijing Municipal Natural Science Foundation (No. 4222038), National Key R&D Program of China (2018YFB0803701-1, 2021YFF0307602) and Fundamental Research Funds for the Central Universities (CUC210A003). This research was funded partly by Research Foundation of School of Computer and Cyberspace Security (Communication University of China), partly by Open Research Project of the State Key Laboratory of Media Convergence and Communication (Communication University of China), partly by the National Key R&D Program of China, and partly by Fundamental Research Funds for the Central Universities.

References

1. Mansour, M.F., Subspace Design of Compactly Supported Orthonormal Wavelets. Journal of Fourier Analysis and Applications, 2014. 20(1): p. 66-90.
2. Peeters, R. and A.J.E. Karel, Data driven design of an orthogonal wavelet with vanishing moments, in 21st International Symposium on Mathematical Theory of Networks and Systems. 2014.
3. Han, B. and H. Ji, Compactly supported orthonormal complex wavelets with dilation 4 and symmetry. Applied and Computational Harmonic Analysis, 2009. 26(3): p. 422-431.
4. Toda, H. and Z. Zhang. A new type of orthonormal wavelet basis having customizable frequency bands. 2015: IEEE.
5. Yanbo, L. and W. Liancheng, The Construction of Orthogonal Basis of Compact Supported Wavelets. Journal of Minzu University of China (natural science edition), 2009. 18(suppl.).
6. Fu Qinyi, J.S., Construction of a compactly supported biorthogonal wavelet basis. Journal of Vibration Engineering, 2004. 17(3).
7. Mansour, M.F. On the design of matched orthonormal wavelets with compact support. 2011: IEEE.
8. Mansour, M.F. On the design of orthonormal wavelets for finite-length signals. 2013: IEEE.
9. Karel, J. and R. Peeters, Orthogonal Matched Wavelets with Vanishing Moments: A Sparsity Design Approach. Circuits, systems, and signal processing, 2018. 37(8): p. 3487-3514.
10. He, T. and T. Nguyen, A Note on the Daubechies Approach in the Construction of Spline Type Orthogonal Scaling Functions. 2015.
11. Nguyen, T., Construction of Spline Type Orthogonal Scaling Functions and Wavelets. Honors Projects, 2015.
12. Yang, S. and H. Huang, A novel method of constructing compactly supported orthogonal scaling functions from splines. Journal of Inequalities and Applications, 2017. 2017(1).
13. Gupta, K.L., B. Kunwar and V.K. Singh, Compactly Supported B-spline Wavelets with Orthonormal Scaling Functions. Asian Research Journal of Mathematics, 2017.
14. Qin, P., et al., Convolutional neural networks and hash learning for feature extraction and of fast retrieval of pulmonary nodules. Computer Science and Information Systems, 2018. 15(3): p. 517-531.
15. Wu, J., et al., Click-boosted graph ranking for image retrieval. Computer Science and Information Systems, 2017. 14(3): p. 629-641.
16. Zheng, H., et al., A novel framework for Automatic Chinese Question Generation based on multi-feature neural network model. Computer Science and Information Systems, 2018. 15(3): p. 487-499.
17. Fan, Y., et al., One enhanced secure access scheme for outsourced data. Information sciences, 2021. 561: p. 230-242.

18. Fan, Y., et al., Privacy preserving based logistic regression on big data. Journal of network and computer applications, 2020. 171: p. 102769.
19. Fan, Y., et al., A secure privacy preserving deduplication scheme for cloud computing. Future Generation Computer Systems, 2019. 101: p. 127-135.
20. Fan, Y., et al., Fine-grained access control based on Trusted Execution Environment. Future Generation Computer Systems, 2020. 109: p. 551-561.
21. Fan, Y., et al., One secure data integrity verification scheme for cloud storage. Future generation computer systems, 2019. 96: p. 376-385.
22. Fan, Y., et al., SBBS: A Secure Blockchain-based Scheme for IoT Data Credibility in Fog Environment. IEEE Internet of Things Journal: p. 1-1.
23. Liang, W., et al., Secure Data Storage and Recovery in Industrial Blockchain Network Environments. IEEE Transactions on Industrial Informatics, 2020. 16(10): p. 6543-6552.
24. Fan, Y., et al., TraceChain: A blockchain‐based scheme to protect data confidentiality and traceability. Software: Practice and Experience, 2019. 77.
25. Zhang Bin, Y.F., Wavelet analysis method and its Application. 2011, Bei jing: National Defense Industry Press.
26. Youxin, L., H. Zheming and C. Xiaoyi, The least square method of hyperchaotic sequence for solving nonlinear equations and its application. Journal of Hunan University of Arts and Sciences (natural science edition), 2010(22).
27. Li, G., Numerical optimization method. 2014, Beijing: Peking University Press.

Yongkai Fan received the Bachelor, Master and Ph.D. degrees from Jilin University, Changchun, China, in 2001, 2003, and 2006. His current appointment is an associate professor at the Communication University of China, and his current research interests include theories of software engineering and software security.

Qian Hu received the B.S. degree in Mathematics and Applied Mathematics from Communication University of China, in 2016. And now she is pursuing for a master degree of Electronic Information in Communication University of China. Her current research interests include explainable AI.

Yun Pan received the B.S. degree from Northwest Normal University, in 1995, the M.S. degree from Liaoning Petrochemical University University, in 2001, and the Ph.D. degree from China University of Mining and Technology-Beijing, in 2003, all in engineering. She is currently a Professor with the Communication University of China. Her current research interests include network security, blockchain, and the future Internet architecture.

Chaosheng Huang received the Bachelor, Master and Ph.D. degrees from Jilin University, Changchun, China, in 1992, 1999, and 2005. In 2019, he joined Tsinghua University and his current research interests include development methodology of intelligent connected vehicle and safety of the intended functionality.

Chen Chao received the Bachelor degree from Beijing Institute of Technology in 2004, and the Master and Ph.D. degrees from Communication University of China in 2006 and 2010. He was awarded the title of Associate Researcher in 2013, and his current research directions are communication engineering, signal processing, and broadcasting and television engineering.

Kuan-Ching Li is currently a Distinguished Professor at Providence University, Taiwan. He is a recipient of awards and funding support from several agencies and industrial companies, as also received distinguished chair professorships from universities in China and other countries. Besides publication in refereed journals and top conferences papers, he is co-author/co-editor of several technical professional books published by CRC Press/Taylor & Francis, Springer, and

McGraw-Hill. Dr. Li's research interests include GPU/manycore computing, Big Data, and cloud. He is a senior member of the IEEE and a fellow of the IET.

Weiguo Lin is a professor at the Communication University of China. His research interests include digital rights management and information security technologies in broadcasting and converging media applications. He received a Ph.D. in communications and information systems from the Communication University of China in 2011.

Xingang Wu received a bachelor's degree and graduated from Tianjin Trade Union Management Cadre College in 2007, he is a software engineer in the school of vehicle and transportation of Tsinghua University. His current research interests include software engineering, intelligent vehicle safety theory and Internet of things security theory.

Yaxuan Li major in Information Security. And she now is studying in School of Computer and Cyber Science from Communication University of China, China.

Wenqian Shang received the Bachelor, Master and Ph.D. degrees from Southeast University, Nanjing, China, in 1994, National University of Defense Technology, Changsha, China, in 1999, and Beijing Jiaotong University, Beijing, China, in 2008. Her current appointment is a professor at the Communication University of China. Her current research interests include Artificial Intelligence (AI), Machine Learning and NLP.

Received: April 10, 2021; Accepted: September 20, 2021.

Computer Science and Information Systems 19(2):619–637 https://doi.org/10.2298/CSIS210418065A

Eye Movement Analysis in Simple Visual Tasks

Kiril Alexiev and Teodor Vakarelsky

Institute of Information and Communication Technologies
Bulgarian Academy of Sciences
1113 Sofia, Bulgaria
alexiev@bas.bg
teodor.vakarelsky@gmail.com

Abstract. The small eye movements in the process of fixation on an image element give us knowledge about the human visual information perception. An in-depth analysis of these movements can reveal the influence of personality, mood and mental state of the examined subject on the process of perception. The modern eye tracking technology provides us with the necessary technical means to study these movements. Nevertheless, still a lot of problems remains open. In the present paper two approaches for noise cancellation in the eye-tracker signal and two approaches for microsaccade detection are proposed. The analysis of the obtained results can be a good starting point for interpretation by neurobiologists about the causes of different types of movement and their dependence on the individuality of the observed person and the specific mental and physical condition.

Keywords: fixation eye movements, eye tracker, microsaccade detection.

1. Introduction

The eye movements have long attracted the attention of specialists. They can be used to diagnose a person's mental state, his ability to concentrate, the speed of perception of visual information and much more. In the performance of various visual tasks the eye movement is influenced by external stimuli (the observed scenario), complex cognitive processes and by some unconditional fine sensorimotor mechanism. The so-called eye-trackers (ET) are used to study the eye movement, with the help of which the parameters of the eye movement are obtained. The latest generation of ETs are "non-obtrusive" (can not interfere with a person's freedom of movement) and can provide information with sufficient accuracy even for rapid eye movements. When a subject scans a visual scene two types of the state of his eyes occur: fast movements or saccades and periods of eyes fixations, when only a small amplitude movements are registered. This article discusses the fixational movements only. The fact of existence of fixation of the gaze on certain objects in the scene has been known for a long time. As early as 1879, the French ophthalmologist Emile Javal reported two main conditions of the eyes when a person is reading - rapid jumps (saccades, in fact, the term saccades was first used by Javal, which is in French and it is analogous to the English word jerk) and pauses [31]. Latter, it was discovered that even in pauses (the process of fixation), the eyes continue to move constantly and the described trajectories look chaotic and wandering. The instantaneous speed of these movements is not low, particularly when the head is not specially fixed. It was estimated to

a value of about 1degree per second [4,23,30]. However, due to the ever-changing velocity vector, the average velocity of eye movement is low [10].

Fixational eye movements are subdivided into microtremor, drift and microsaccades [17,9]. It is assumed that these movements lead to small displacements of the image on the fovea when observing a still scene. These movements have not been clearly defined so far and their separation is still a scientific challenge [27]. There is also no well-proven theory about the visual functions of each of these micro movements.

ET is an appropriate measurement equipment for eye movement registration and it is assumed also the presence of Gaussian additive noise in the received signal.

The role of different eye movements in performing fixation has been intensively studied and analyzed for decades, but there are many difficulties in elucidating their physiological and perceptual functions. To this day, there is no consensus on these issues [27,19,26]. While for the larger saccades it has been established with high reliability that they move the image of the observed object in the area of the fovea with the highest resolution, the purpose of the microsaccades, tremor and drift is still a debatable issue.

The **fixation** of the gaze on a specific stationary object is characterized by a relatively stationary projection of the image of this object on the most sensitive part of the fovea. Usually, the fixation is associated with the time between two major saccades, when there are no other big eye movements and there only eye tremor and drift exist. Some scientists claim that the duration of fixation is about $250ms$ [14].

Saccades are relatively large eye movements in the range of $5^0 - 40^0$ [25].The saccade rate is approximately 2-3 times per second [27]. They serve to focus on the observed object by moving it to the most sensitive area of the fovea. Some of them can be considered volitional/conscious. For example, while performing large search saccades, we perform mostly unconscious eye movements during natural vision. We are also able to move to the conscious performance of saccades when performing a specific visual task. Their duration is estimated in the range of 30-80ms (112ms in [25]). In the problem of volitional fixation for a long period of time on a stationary object, considered in the present work, it turned out that the saccades cannot be suppressed all the time and they appear periodically.

Microsaccades are relatively fast eye movements (have the same dynamic characteristics as saccades), but with a smaller amplitude (up to 12 angular minutes usually) [9,13] and a frequency of $1 - 2$ times per second [27]. The purpose of the micro-saccades has not yet been definitively clarified and is a moot point. While it was initially thought to be "noise" in the system, in recent years there has been a growing perception that microsaccades have a function of repositioning within the fovea during periods of fixation [27]. Other scientists consider microsaccades as compensatory eye movements for stabilization against head and body movements, which are guided by both visual and vestibular inputs [17]. Microsaccades are thought to be irregular and rare with a frequency of about 1 microsaccade per second [11,17] and their statistics reflect both perceived visual information and changes in a person's cognitive state. The relationship between microsaccades and visual concentration has recently been confirmed by some researchers and they believe that microsaccades have been used to fine-tune the gaze [16]. There are also articles that provide evidence of reducing the number of microsaccades when performing work with high precision - threading or aiming, for example. In the specific experiments conducted in the present work, we assume that the microsaccades are involuntary/unconscious and the observed person does not even know about their existence.

Ocular microtremor is a small (up to one angular minute [27]) high-frequency eye tremor caused by extra-muscular activity stimulated by impulses emitted from the oculomotor area of the brainstem. It is inherent to all people. Several studies have shown that the incidence of this tremor is reduced in patients whose consciousness is decreased by anesthesia or head injury [6]. Ocular tremor is a non-periodic oscillation in the range of 70 - 103 Hz and the average frequency is 83.68 Hz [7].

Eye drift is another smoother and slower movement of the eyes when fixed. The similarity of the movement as a result of eye drift with the brown movement of the particles is often pointed out - randomly and arbitrarily. The frequency of ocular deviations is significantly lower than that of microtremor and is in the range 0 - 40 Hz, and the amplitude of change - about 1.5' - 4' with an average speed of about 4'/s [10]. One of the hypotheses for this eye movement pointed out that the eye drift helps to be obtained higher resolution information for the observed stationary objects [18,2]. In another paper a suggestion was proposed that eye drift is related to the coding and processing of visual information in space and time [1]. All of these fixational eye movements were considered vital for the observation of stationary scenes that were thought to fade over time without movement [24]. Although this remains a controversial topic to this day [27], it is clear that these movements play a very important role in the perception of information by the brain and their analysis can reveal basic properties of the visual system. The correct classification of one or another type of fixational eye movements is a challenge [27], to which this article is devoted.

The small eye movements during fixation is the main object of research in this article. Several algorithms for different fixational eye movement segmentation are proposed. The block diagram of the proposed algorithms for eye tracker data processing is shown on Fig. 1. The obtained results are discussed.

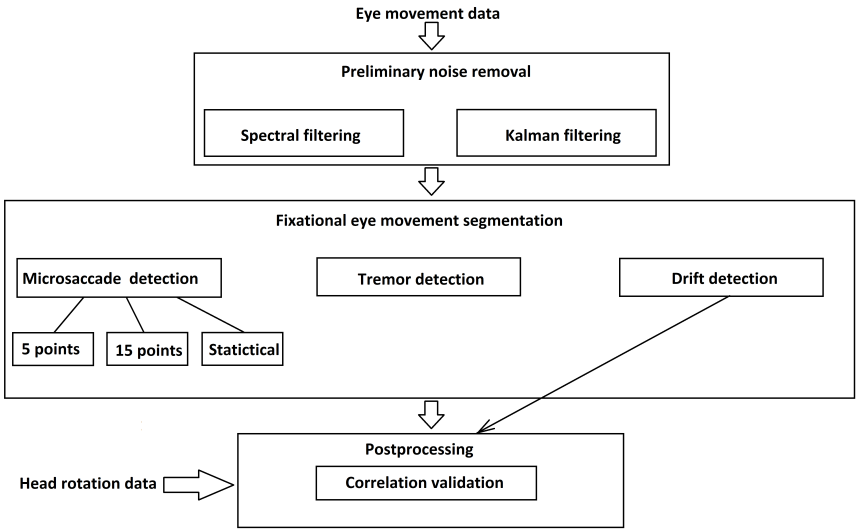


Fig. 1. The block diagram of the proposed algorithms for eye tracking data processing

The paper is organized as follows. After short review of state of the art in the field of fixational eye movement research a mathematical model of fixational eye movement is proposed in Chapter II. Then a description of experimental setup is given. Two noise filtering techniques are proposed in Chapter IV. The first one considers spectral filtering, the second one - Kalman filtering. Chapter V gives description of two methods for microsaccade detection. One of them is based on eye movement velocity, the other analyses the statistical characteristics of ET signal. In Chapter VI microtremor and eye drift are detected and separated. The analysis of results is discussed in Chapter VII and finally the authors contributions are summarized in Conclusion.

2. Eye Movement Description

In this article the fixational eye movements will be commented only. From the brief description in the previous chapter it is clear that there are several types of fixational eye movements with insufficiently clear origin and purpose. Due to the significant differences in frequency and amplitude values of the three types of movements during fixation (microsaccades, microtremor and drift) we will accept the hypothesis that they are additive. Let us denote the movement of the eyes in the process of fixation by m. The movement of the eyes is carried out by rotating the eyes in horizontal and vertical directions. Therefore, the movement of the eyes in the process of fixation will be described by the two-dimensional vector $m = \begin{bmatrix} m^{(h)} \\ m^{(v)} \end{bmatrix}$ and will be measured by the degrees of rotation [deg] on each of its components (h - horizontal and v - vertical). In clinically healthy observed individuals, we will assume that $m = \frac{m_l + m_r}{2}$, where m_l is the rotation magnitude vector of the left eye and m_r is the rotation magnitude vector of the right eye. In fixation mode, according to our hypothesis:

$$m = m_s + m_t + m_d \,, \tag{1}$$

i.e. the total eye movement is equal to the sum of the rotations caused by microsaccades, microtremor, and eye drift. The eye tracker, eye rotation measuring instrument, used in the experiments, measures the values of the components of the vector m every $0.001s$. Like any measuring device, eye tracker measures the rotation of the eyes with an error that with high plausibility could be considered as Gaussian distributed and additive to the measured values:

$$m^{(ET)} = \begin{bmatrix} m^{(ET_h)} \\ m^{(ET_v)} \end{bmatrix} = \begin{bmatrix} m^{(h)} \\ m^{(v)} \end{bmatrix} + \begin{bmatrix} \epsilon^{(h)} \\ \epsilon^{(v)} \end{bmatrix} = \begin{bmatrix} m_s^{(h)} + m_t^{(h)} + m_d^{(h)} + \epsilon^{(h)} \\ m_s^{(v)} + m_t^{(v)} + m_d^{(v)} + \epsilon^{(v)} \end{bmatrix} , \tag{2}$$

Eliminating noise and separating the different types of movements in the scan path is a mandatory step in the analysis of eye movement. The quality of the subsequent analysis also depends on the quality of noise reduction and fixational eye movement detection and separation. In the following chapters several algorithms for noise removal and eye movement separation are proposed and analyzed.

3. Experimental Setup

The eye movements of the participants in the experiment were recorded by a specialized hardware - "Jazz novo" eye tracking system (Ober Consuting Sp. Z o.o.). The recordings from all the sensors of the device for one session per person were collected with 1 kHz frequency rate and the information is saved in files for analysis. The obtained sensor data include: the calibration information; records of horizontal and vertical eye positions in degrees of visual angle; information received from device accelerometers (vertical and horizontal accelerations) and gyroscopes (rotation rate in vertical and horizontal planes); information about tested subjects and type of the experimental trial for each particular record.

The eye tracker "Jazz novo" is a mobile nonobtrusive device, which consists of several sensors:

- Monocular eye tracking sensor which measures the rotations of both eyes and gives as an output an averaged rotation. The sensor is located above the person's nose;
- 2-axial gyroscope, which measures the speed of rotation rate of the head in Y and Z (yaw) axes;
- 2-axial accelerometer, which tracks the acceleration of the head in Y (pitch) and Z (yaw) axes.

The stimuli were presented on a gray screen with mean luminance $50cd/m^2$ using $20.1''$ NEC MultiSync LCD monitor with Nvidia Quadro 900XGL graphic board at a refresh rate of 60 Hz and screen resolution 1280×1024 pixels.

The stimulus, which the test subjects were exposed to, consists of PowerPoint animation, shown on a dark background, displayed on a 41x31 cm screen. The test subject is sitting at a distance 58 cm from display. Scenario consists of several tasks: eye fixation on a dot; tracking a moving dot on the screen (dynamical task); several search tasks on a complex scenario. The first part of experiment is discussed in this article.

The especially designed scenario includes five consecutively appearing dots, located in the middle of the left display edge, in the middle of the right display edge, in the middle of the upper display edge, in the middle of the lower display edge and in the middle of the screen, respectively. Every of these points remains stationary $15s$ and then disappears.

Eight persons were examined. Six of them were of age between 20 and 30 years. One participant was 40 years old and the last one was 60 years old. The results from one of them are presented in the article, the results from others were used to tune and validate algorithms for eye tracker data processing.

4. Eye Tracker Signal Filtering

Signal noise filtering aims to cut or to attenuate the noise and its effect on the useful signal. The ET signal for horizontal eye movement and 2D eye movement in fixational state are displayed on Fig. 2 and Fig. 3 correspondingly. Two approaches to noise filtration have been implemented.

The first of these is based on our a priori knowledge of the frequency properties of eye movement during fixation on an object. It has been found that the highest frequency component of these movements is of the order of 104 Hz. The first noise suppression

algorithm converts the ET signals $m^{(ET_h)}$ and $m^{(ET_v)}$ from time domain into frequency domain and reset the frequency coefficients higher than 104 Hz to zero. Then the signal is converted from the spectral to the time domain back. The results of each of the filtration steps are shown on Fig. 4 and Fig. 5:

Additionally several peaks were found in the regions around 50 Hz, 100 Hz and 150 Hz. We consider that there are harmonics of power supply with standard frequency of 50 Hz. The region around 50 Hz ($\pm$ 0.01 Hz) has been zeroed in order to remove the power supply influence.

Clearing certain frequencies gives more realistic picture of the real eye movement. Most of the predictable factors have been cleared. Some random body movement and unpredictable sensor disturbances still remain uncleaned.

The second filtering approach uses a two-dimensional Kalman filter [5]. We choose the window size of 11 measurements ($10 * 0.001s = 0.01ms$ - the window size corresponds to a frequency of 100 Hz). In this window the statistical characteristics of the signal are calculated. It is assumed that these are the statistical characteristics of the additive noise (frequencies are 100 Hz and higher).

The state equation is defined as:

$$s_{k+1} = Fs_k + \omega_k \tag{3}$$

In this equation s_{k+1} is the state vector at the $k+1$ -st moment and $s_{k+1} = \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \end{bmatrix}$, where $x, \dot{x}$ are the horizontal location of the eye and its rotation speed in degrees and degrees per second, respectively, and $y, \dot{y}$ are the vertical location of the eye and its rotation speed. The matrix $F = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}$ is a matrix of the transition from state in k-th moment to state in k+1-th moment.The vector $\omega_k = \begin{bmatrix} \sigma_x^2 \\ \sigma_{\dot{x}}^2 \\ \sigma_y^2 \\ \sigma_{\dot{y}}^2 \end{bmatrix}$ sets the uncertainty in the change of the state vector (the uncertainty in the model of the observed system). It is assumed the uncertainty to have multivariate normal distribution with zero mean and covariance Q_k: $\omega_k \sim N(0, Q_k)$. The covariance matrix of the error of the state vector at the k-th moment is expressed by:

$$Q_k = \begin{bmatrix} \sigma_{xp}^2 1/3\Delta t^3 & \sigma_{xp}^2 1/2\Delta t^2 & 0 & 0 \\ \sigma_{xp}^2 1/2\Delta t^2 & \sigma_{xp}^2 \Delta t & 0 & 0 \\ 0 & 0 & \sigma_{yp}^2 1/3\Delta t^3 & \sigma_{yp}^2 1/2\Delta t^2 \\ 0 & 0 & \sigma_{yp}^2 1/2\Delta t^2 & \sigma_{yp}^2 \Delta t \end{bmatrix}$$

The equation of the relationship between the measurement vector and the state vector looks like this:

$$z_k = Hs_k + v_k, \tag{4}$$

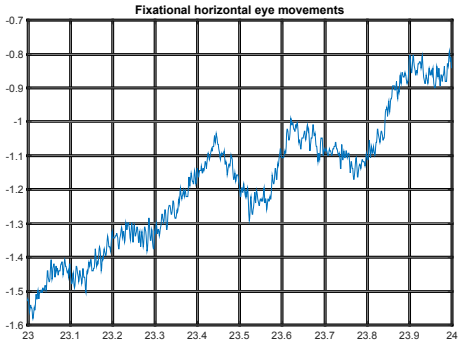


Fig. 2. ET noisy signal for horizontal eye movement

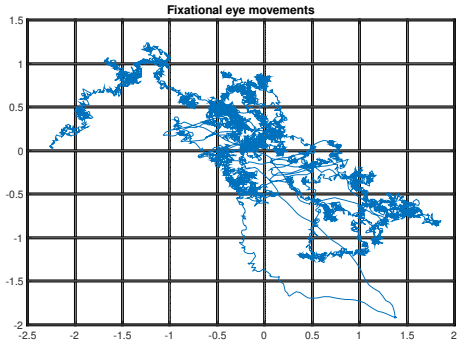


Fig. 3. 2D ET signal (for 2D eye movement)

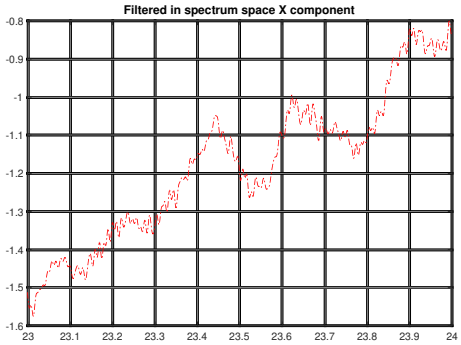


Fig. 4. Filtered in frequency domain ET measurements (horizontal only movement)

where $z_k = \begin{bmatrix} z_x(k) \\ z_y(k) \end{bmatrix}$ is a vector of measurements obtained by ET for eye movements in the horizontal and vertical directions, respectively; $H = \begin{bmatrix} 1\,0\,0\,0 \\ 0\,0\,1\,0 \end{bmatrix}$ is the correspondence

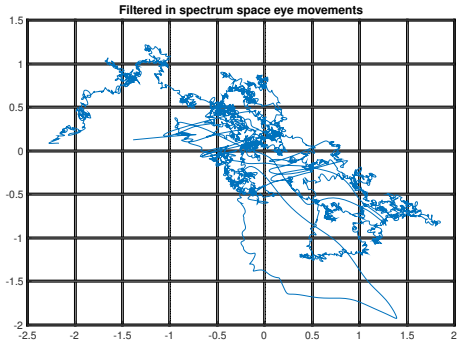


Fig. 5. Filtered in frequency domain ET measurements (2D movement)

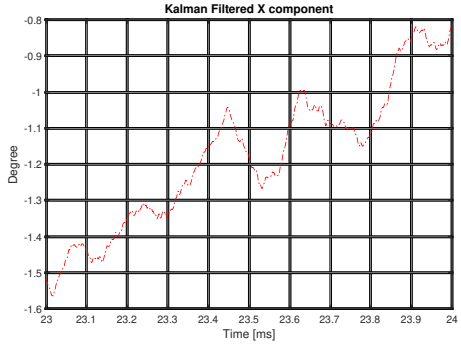


Fig. 6. Kalman filtered measurements (horizontal eye movement)

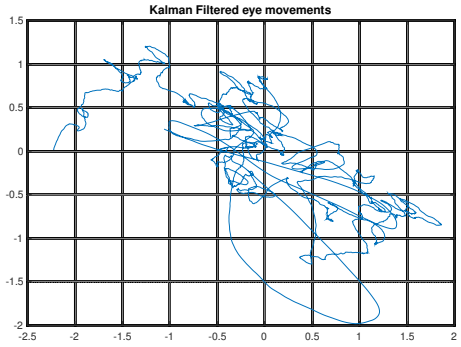


Fig. 7. Kalman filtered measurements (2D eye movement)

matrix between the state vector and the vector of the measurements and $R_k = \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}$ is the covariance matrix of the errors of the used ET $(v_k \sim N(0, R_k))$.

The Kalman filter is described by the following two main steps:

Prediction step:

$$s_{k+1/k} = F\hat{s_k}, \tag{5}$$

Here $s_{k+1/k}$ is the predicted state of the system at $k+1$-st moment based on the estimated state vector s_k at the previous k-th moment.

$$P_{k+1/k} = FP_{k/k}F^T + Q_k, \tag{6}$$

where $P_{k+1/k}$ is the predicted value of the covariance matrix of the error based on the specified covariance matrix of the error $P_{k/k}$ at the k-th moment and the covariance matrix of the error of the state vector Q_k.

Update step:

Finding the innovation for the predicted value:

$$\nu_k = z_k - Hs_{k/k-1} \tag{7}$$

Calculation of the gain:

$$K_k = P_{k/k-1}H^T(R + HP_{k/k-1}H^T)^{-1} \tag{8}$$

Finding the updated status vector:

$$s_{k/k} = s_{k/k-1} + K_k\nu_k \tag{9}$$

Finding the updated error covariance matrix:

$$P_{k/k} = (I - K_kH)P_{k/k-1} \tag{10}$$

Fig. 6 and Fig. 7 show the results of applying the Kalman filter on the measurements obtained from ET. The received results are for the following values of Kalman filter parameters: $\Delta t = 0.001s, \sigma_{xp}^2 = \sigma_{yp}^2 = 10000, \sigma_x = 3.35, \sigma_y = 2.67$ [3].

The proposed filters for noise rejection have some principal differences. The spectral filtering algorithm is based on the knowledge of the useful spectrum of data obtained from eye tracker and removes everything outside this spectrum. In the case of noise presence in the bandpass region, this noise is not removed and will continue to affect the useful signal. The electricity supply $(50Hz)$ should serve as an example of noise in the allowable spectrum range. Special measures have been taken to be eliminated.

The second method uses the analysis of the eye tracker data [3]. Applying Kalman filter with parameters, corresponding to the used apparathus the eye tracker measurement error will be minimized, but the part of the useful signal will be affected also and this leads to deviations in the results.

Due to the difficulty of making an accurate balance of which of the methods is better, both approaches are presented and depending on the problem to be solved and the equipment used, the more appropriate one can be chosen.

5. Microsaccade Detection

Microsaccades are relatively large movements of the eye in fixation mode. If we assume that the microsaccade detecting technique is similar to the technique of saccade detection, then the following methods could be applied [28]:

- Methods based on the speed of movement of the eye. These methods use the hypothesis of a higher velocity of the eye in saccade compared with other eye movements in fixation mode. The same hypothesis could be successfully applied to distinguish microsaccades from tremor and drift;
- Methods using dispersion. These methods are based on the dispersion, which in this case is a measure of the deviation of the gaze from the point of fixation. The method is also applicable to distinguish microsaccades from drift and ocular tremor, due to the different statistics of these movements;
- Methods using the fixation area. The article cited above also mentions so-called microfixations. Under microfixations the authors considered fixation points, which are closely located and the movements in them could be performed by the microsaccades we are looking for;
- Methods using the saccade duration. These methods use the predefined information that the fixation is seldom shorter than 100 ms, most often in the range between 200-400 ms;
- Locally adaptive methods. This type of method estimates the approximate duration of fixation of each observed person and uses it for subsequent signal analysis. Therefore, this method is adaptive to the specifically observed person.

In this article we implemented two algorithms for microsaccade detection. The microsaccade is defined as continuous eye movement, exceeding a certain velocity threshold.

The first of them is based on finding the highest speeds of eye movement during fixation (Fig. 8). To reduce the impact of outliers, eye movement velocities are calculated using windows with length of 5 or 15 points. This part of the algorithm can be regarded similar to the classical algorithm of Engbert and Kliegl ([12]), but with the use of other window sizes. Finding the maxima in the (average) 2D velocities of eye movement, the search for the starting and end points of each microsaccade begins. Unlike the Otero-Millan et al. ([22]) algorithm, where the starting points are determined by means of a constant threshold ($3^0/s$) in the proposed algorithm an adaptive threshold is chosen depending on the average speed of eye movement during fixation. A term of a reasonable distance for connecting neighboring microsaccades is also introduced. It serves to merge saccades if the distance between them is smaller than a threshold. The obtained results of saccade detection are shown on Fig. 8, Fig. 9, Fig. 10 and Fig. 11 for two cases of 5 and 15 points windows.

The second solution is based on statistical signal analysis. We assume that the statistical signal parameters change significantly during microsaccade motion. The algorithm for microsaccade detection looks for an event of a significant change in the dispersion of the signal received from ET in the fixation mode. In order to test this hypothesis a window signal processing is organized, considering 11 consecutive measurements. For the current window, the obtained measurements can be considered as a realization of 11 normally distributed random variables. They can also be normalized to the sum of their squares χ^2_{10}:

$$\chi^2_{10} = \Sigma^{11}_{i=1}((X_i - \mu_i)/\sigma_i)^2$$

which represents a chi-square distribution with 10 degrees of freedom. Here X_i denotes the i-th measurement in the window, $\mu_i = (\Sigma^n_{i=1} X_i)/n$ is the mean of the measurements

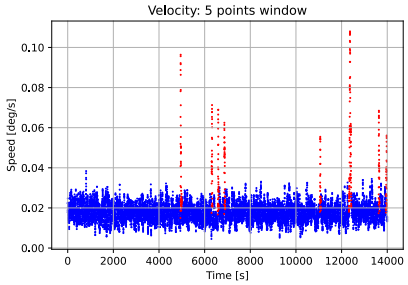


Fig. 8. Velocity microsaccade detection (5 points window; blue - horizontal fixational eye movement; red - microsaccades)

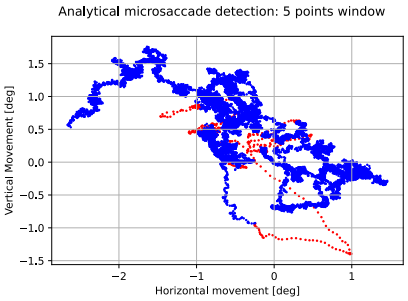


Fig. 9. Velocity microsaccade detection (5 points window; blue - 2D fixational eye movement; red - microsaccades)

in the window, and σ_i is the variance of the measurements in the window. The following hypothesis will be tested:

$$H_0 : \sigma^2 = \sigma_0^2$$
$$H_1 : \sigma^2 \neq \sigma_0^2$$

To test the hypotheses we use a significance level $\alpha = 0.01$. The obtained results are shown in Fig. 12. The statistical microsaccade detector finds more segments in comparison with velocity microsaccade detector (see Fig. 8, Fig. 10 and Fig. 12). Nevertheless the contours of the detected microsaccades often remain torn is several subsegments (Fig. 13). Summarizing the received results, it becomes obvious that the proposed microsaccade detectors complement each other, which leads to the conclusion that in the microsaccade detectors it should be applied a more complex criterion, and not just a single restriction.

6. Eye Tremor and Drift Detection and Segmentation

The eye tremor and drift segmentation could be fulfilled by frequency filtering. The source signal is ET Kalman filtered signal between second and third microsaccades (see Fig. 12). Applying bandpass filters for the frequencies bands 0-40 Hz and 70-104 Hz the corresponding eye drift (Fig. 14) and tremor (Fig. 15) are segmented .

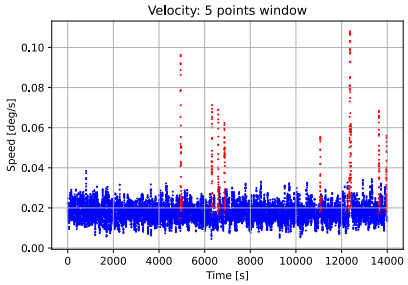


Fig. 10. Velocity microsaccade detection (15 points window; blue - horizontal fixational eye movement; red - microsaccades)

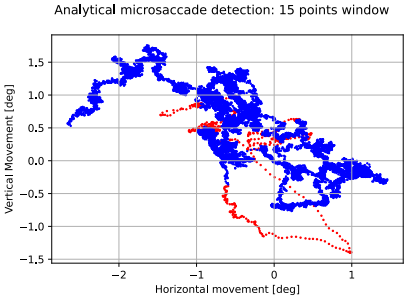


Fig. 11. Velocity microsaccade detection (15 points window; blue - 2D fixational eye movement; red - microsaccades)

7. Analysis of the Received Results

The obtained results demonstrate the abilities of the proposed algorithms to eliminate the influence of power supply on the readings of ET. Applying different filters minimizes the influence of additive high-frequency Gaussian noise. The optimal setting of the parameters of these filters is a prerequisite for obtaining a quality result achieving maximum noise reduction without loss of useful information. Both proposed noise rejection algorithms successfully reduced additive noise. Below we compare the characteristics of the Kalman filter with those of the "classic" Engbert - Kliegl filter [12] (based on the five points average $\overrightarrow{v}_n = (\overrightarrow{x}_{n+2} + \overrightarrow{x}_{n+1} - \overrightarrow{x}_{n-1} - \overrightarrow{x}_{n-2}/6\Delta t)$ and its 11 - point modification in [29] (where $\overrightarrow{v}_n = [\sum_{i=1}^{5}(\overrightarrow{x}_{n+i} - \overrightarrow{x}_{n-i})]/30\Delta t$). The Engbert - Kliegl filter and its modification were realized by implementation of linear convolution filters with corresponding kernels.

To compare denoising algorithms one and same signal is passed through the three filters. The signal was taken from https://github.com/sheynikh/msdetect, quoted in [29]. The chosen segment (in the interval $48s - 58s$) from the whole signal is is very close to the signal, presented on fig. 2A of the same article. The signals after denoising filters are depicted on Fig. 16. Finally, the signal-to-noise ratio is calculated for the visualized segment of signal with microsacades, labeled in advance. For Kalman filtered signal

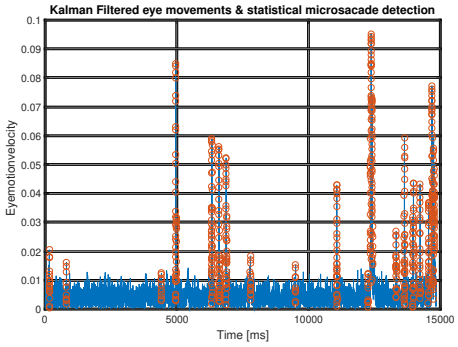


Fig. 12. Statistical microsaccade detection for Kalman filtered signal: blue - horizontal fixational eye movement; red - microsaccadic eye movement)

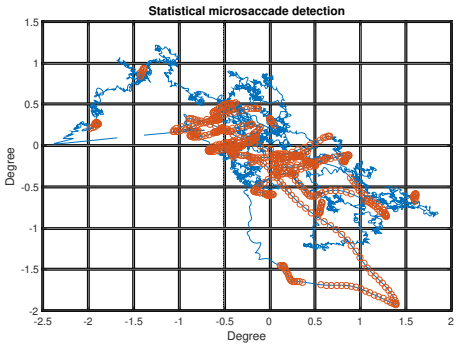


Fig. 13. Statistical microsaccade detection for Kalman filtered signal: blue - 2D fixational eye movement; red - microsaccadic eye movement)

$SNR_{Kalman} = 20.15dB$; for E&K filter $SNR_{E\&K} = 14.38dB$ and for modified (by Sheynikhovich et al.) filtered signal $SNR_{S\&etc} = 18.10dB$. The best results of Kalman filtering are received without any losses in microsaccades (Fig. 16).

The detection of microsaccades in a state of fixation, however, is burdensome. This is due to the uncertainty in the definition of microsaccades. Two different algorithms with window technique for detecting microsaccades were proposed in the article. In the first of these approaches we are looking for eye movements performed at a higher speed than normal in the fixation process. Two windows of 5 and 15 consecutive measurements were used. Microsaccade detector with longer window allows more precise segmentation of the fixational microsaccades. However, in the 2D image of fixational eye movement (see Fig. 9 and Fig. 11), some trajectories can be distinguished that strongly resemble saccades, but have a lower (close to normal) speed of eye movement. These examples prove that the criterion for detecting saccades by the speed of eye movement is not precise enough.

In the next experiment we examine the differences in the detection of microsaccades by four different algorithms: the newly proposed 5-point window, 15-point window and statistical algorithms, described in this paper, and the algorithm of Sheynikhovich at al.

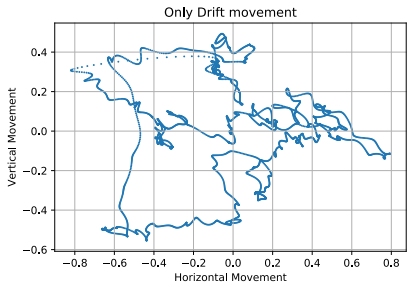


Fig. 14. Isolated eye drift

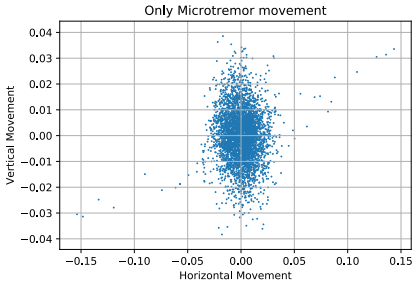


Fig. 15. Isolated eye tremor

in [29]. The microsaccade detectors were tested on the same segment of data from the previous experiment. The number of labeled saccades is 15 (according to the public data and software, cited in[29]). The summary of results are presented in the Table 1:

Table 1. Microsaccade detection

Algorithm	**Original signal**	**Noised signal** $\sigma = 0.005$	**Noised signal** $\sigma = 0.02$
Labeled microsaccades	15	15	15
Sheynikhovich et al.	15	15	15
5-points window	19 (4 additional)	17 (2 additional)	17 (2 additional)
15-points window	17/18 (3 additional; 1 merged)	16/17 (2 additional; 1 merged)	16/17 (2 additional; 1 merged)
Statistical	19 (4 additional)	19 (4 additional)	17 (2 additional)

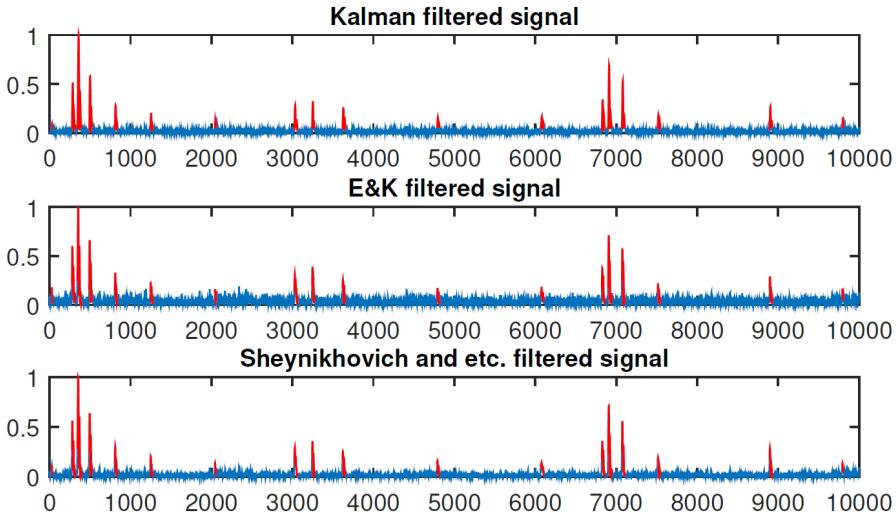


Fig. 16. Comparison between Kalman filter, Engbert - Kliegl filter and Sheynikhovich filter

All tested algorithms work well with denoised signals or signals with weak noise. They all detected the micro-saccades marked by experts. The algorithms working with a five point window and the statistical algorithm detected four additional microsaccades. The algorithm working with a 15 point window detected only three out of the four additional microsaccades. These (four) saccades are marked in red in the 2D representation of eye movement (see Fig. 17). There is no doubt, for at least two of them, that these eye movements are microsaccades. The non-registration of these movements by experts requires additional commentary. It is probably due to the velocities of eye movements close to the thresholds for detection of microsaccades (statistical characteristics are almost the same as those of non-saccade eye movements - this fact was also established by the statistical detector of microsaccades). It is also interesting to denote, that the 15-point algorithm also detected three of these eye movements as microsaccades, but its larger window inflicts on nearby microsaccades to be merged.

In the next experiment a Gaussian noise with zero mean and standard deviation 0.005 was added to the signal. All the algorithms perform well microsaccade detection. The statistical microsaccade detection algorithm found the same number of microsaccades. The 5 point window algorithm found 17 microsaccades - the 15 labeled saccades and the two not labeled microsaccades. The 15 point window algorithm detected also 17 saccades, but one of them is merged to another. The statistical algorithm found again the same number of 19 saccades.

The high-noise regime was obtained with additive Gaussian noise with zero mean and standard deviation 0.02. In both of the experiments with additional noise was used a Kalman filter for noise reduction. The five and 15 points window algorithms found the same saccades as in the previous paragraph. The statistical algorithm found 17 saccades. Its quality was deteriorated because of high rate of the noise.

During all the experiments the Schenikhovich algorithm found exactly the 15 labeled saccades, showing good noise stability, but it missed to detect the additional (unlabeled) microsaccades.

The used experimental equipment did not contain high precision eye tracker like cited in [15,8,20,21]. There was no possibility to accurately assess the exact position of the image of the object under attention on the retina in time of microsaccades and local image movement. We had no enough information to find evidence of a direct relationship between the microsaccades and accurate fixation or image stabilization on retina.

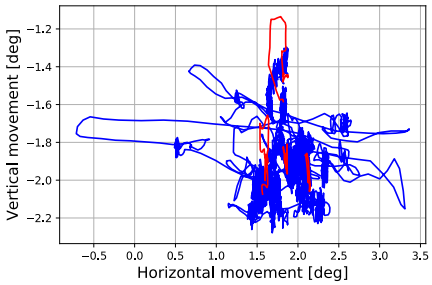


Fig. 17. Microsaccades additionally found

8. Analysis of Fixational Eye Drift

The corrective role of fixational eye drift on image locus was also verified. In the performed experiments the head rotations with respect to the eye movements were estimated. We compare the signals of ET gyros with those of segmented eye drift. The choice to use gyroscope readings is not accidental. Gyroscopes measure the speed of rotation of the head and even a slight rotation of the head leads to a large displacement of the image on the retina and requires correction by the oculomotor complex of the eyes for stabilization of the image on a specific retina area. The accelerometers measure accelerations in the horizontal and vertical directions, and approximately any shaking of the head in the horizontal or vertical direction imparts a corresponding displacement of the image on the retina. The gyroscopes measurements (pitch and yaw velocity of rotation) were processed to estimate exact head rotation. To do this the gyroscope measurements were integrated in order to obtain the angle position of the head. The horizontal drift eye movement and horizontal head movement are shown in Fig. 18 and Fig. 19. An inverse relation between these two movements could be observed. The correlation is estimated to -0.6251.

The given signal on Fig. 18 is disturbed with microsaccadic movements. No dependency between microsaccades and head movements was found. The clear drift signal was isolated by using the microsaccade detection algorithm. All microsaccades were replaced by smooth linear movement. The results are shown on Fig. 19 for the same horizontal drift movement of the eye without any saccades. The correlation is estimated to -0.6332.

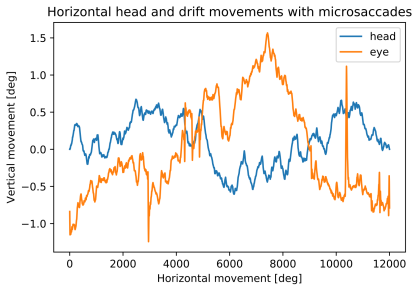


Fig. 18. Correlation between eye and head movements (with microsaccades)

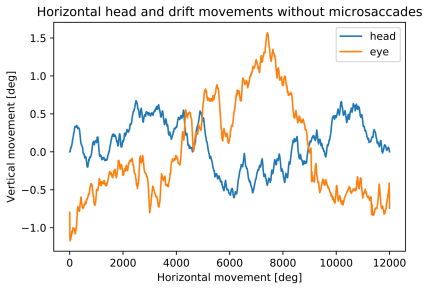


Fig. 19. Correlation between eye and head movements (without microsaccades)

The provided results prove the relation between the drift and the head movement. Something more, a time delay of eye reaction to the head movements is noticed and it can be estimated.

9. Conclusion

This article presents eye movement analysis in simple visual tasks. It considers small fixational eye movements only. A general additive mathematical model of these eye movements is proposed. Two approaches for elimination/reduction of noise in ET measurements and the influence of the power supply of the equipment have been implemented. Two options for detecting microsaccades have also been proposed. The results prove the role of eye drift in image stabilization on the retina of the eye. The analysis shows that new approaches and algorithms for detecting microsaccades should be sought.

References

1. Ahissar, E., Arieli, A.: Figuring space by time. Neuron 32(2), 185–201 (2001)
2. Ahissar, E., Arieli, A.: Seeing via miniature eye movements: A dynamic hypothesis for vision. Frontiers in Computational Neuroscience 6(81), 1–27 (2012)

3. Alexiev, K., Toshkov, T., Dojnow, P.: Enhancing accuracy and precision of eye tracker by head movement compensation and calibration. In: CompSysTech '19: Proceedings of the 20th International Conference on Computer Systems and Technologies. pp. 226–233. ACM, Ruse, Bulgaria (2019)
4. Aytekin, M., Victor, J., Rucci, M.: The visual input to the retina during natural head-free fixation. The Journal of Neuroscience 34(38), 12701–12715 (2014)
5. BarShalom, Y., Li, X., Kirubarajan, T.: Estimation with applications to tracking and navigation. USA (2001)
6. Bojanic, S., Simpson, T., Bolger, C.: Ocular microtremor: a tool for measuring the depth of anesthesia. British Journal of Anaesthesia 86(4), 519–522 (2001)
7. Bolger, C., Bojanic, S., Sheahan, N.F., Coakley, D., Malone, J.F.: Dominant frequency content of ocular microtremor from normal subjects. Vision Research 39(11), 1911–1915 (1999)
8. Bowers, N.R., Boehm, A.E., Roorda, A.: The effects of fixational tremor on the retinal image. Journal of Vision 19(11), 1–16 (2019)
9. Collewijn, H., Kowler, E.: The significance of microsaccades for vision and oculomotor control. Journal of Vision 8(14), 1–21 (2008)
10. Ditchburn, R.W.: Eye-Movements and Visual Perception. Clarendon (Oxford University Press), New York, USA (1973)
11. Engbert, R.: Microsaccades: A microcosm for research on oculomotor control, attention, and visual perception. In: Martinez-Conde, S., Macknik, S., Martinez, L., Alonso, J.M., Tse, P. (eds.) Visual Perception Fundamentals of Vision: Low and Mid-Level Processes in Perception, Progress in Brain Research, vol. 154 Part A, pp. 177–192. Elsevier B.V. (2021)
12. Engbert, R., Kliegl, R.: Microsaccades uncover the orientation of covert attention. Journal of Vision 43(9), 1035–1045 (2003)
13. Hafed, Z.: Mechanisms for generating and compensating for the smallest possible saccades. European Journal of Neuroscience 33(11), 2101–2113 (2011)
14. Kasprowski, P., Ober, J.: Eye movements in biometrics. In: Maltoni, D., Jain, A. (eds.) Biometric Authentication, Lecture Notes in Computer Science, vol. 3087, pp. 248–258. Springer, Berlin, Heidelberg (2004)
15. Hee-kyoung Ko, H.K., Snodderly, D.M., Poletti, M.: Eye movements between saccades: Measuring ocular drift and tremor. Vision Research 122, 93–104 (2016)
16. Ko, H., Poletti, M., Rucci, M.: Microsaccades precisely relocate gaze in a high visual acuity task. Nature Neuroscience 13(12), 1549–1553 (2010)
17. Kowler, E.: Eye movements: The past 25 years. Vision Research 51(13), 1457–1483 (2011)
18. Kuang, X., Poletti, M., Victor, J.D., Rucci, M.: Temporal encoding of spatial information during active visual fixation. Current Biology 22(6), 510–514 (2012)
19. Martinez-Conde, S., Macknik, S., Hubel, D.: The role of fixational eye movements in visual perception. Nature reviews neuroscience (3), 229–240 (2004)
20. Niehorster, D.C., Zemblys, R., Beelders, T., Holmqvist, K.: Characterizing gaze position signals and synthesizing noise during fixations in eye-tracking data. Behavior Research Methods 53, 2515–2534 (2020)
21. Niehorster, D.C., Zemblys, R., Holmqvist, K.: Cis apparent fixational drift in eye-tracking data due to filters or eyeball rotation? Behavior Research Methods 53, 311–324 (2021)
22. Otero-Millan, J., Castro, J., Macknik, S., Martinez-Conde, S.: Unsupervised clustering method to detect microsaccades. Journal of Vision 14(2), 1–17 (2014)
23. Poletti, M., Aytekin, M., Rucci, M.: Head-eye coordination at a microscopic scale. Current Biology (25), 3253–3259 (2015)
24. Riggs, L.A., Ratliff, F., Cornsweet, J.C., Cornsweet, T.N.: The disappearance of steadily fixated visual test objects. Journal of the Optical Society of America 43(6), 495–501 (1953)
25. Robinson, D.A.: The mechanics of human saccadic eye movement. Journal of Physiology 174(2), 245–264 (1964)

26. Rucci, M., McGraw, P., Krauzlis, R.: Fixational eye movements and perception. Vision Research 118, 1–4 (2016)
27. Rucci, M., Poletti, M.: Control and functions of fixational eye movements. Annual Review of Vision Science (1), 499–518 (2015)
28. Salvucci, D.D., Goldberg, J.H.: Identifying fixations and saccades in eye-tracking protocols. In: ETRA '00:Proceedings of the 2000 Symposium on Eye Tracking Research & Applications. pp. 71–78. ACM, Palm Beach Gardens Florida USA (2000)
29. Sheynikhovich, D., Becu, M., Wu, C., Arleo, A.: Unsupervised detection of microsaccades in a high-noise regime. Journal of Vision 18(6), 1–16 (2018)
30. Skavenski, A., Hansen, R., Steinman, R., Winterson, B.: Quality of retinal image stabilization during small natural and artificial body rotations in man. Vision Research 19(6), 675–683 (1979)
31. Wade, N.J.: Pioneers of eye movement research. i-Perception 1(2), 33–68 (2010)

Kiril Alexiev received his M.S. degree from Kyiv Polytechnic Institute (today National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute") and PhD degree from Bulgarian Academy of Sciences. He is currently Head of department Information Technologies for Sensor Data Processing in the Institute of Information and Communication Technologies – Bulgarian Academy of Sciences. His research interests include sensor data processing, data and information fusion, image processing.

Teodor Vakarelsky received BSc degree in Mathematics from Sofia University "St. Kliment Ohridski" in 2020. Currently he is working toward Master of Mathematics in "Computational Mathematics and Mathematical Modeling". He is a member of the research team of department Information Technologies for Sensor Data Processing in the Institute of Information and Communication Technologies - Bulgaria Academy of Science since 2020.

Received: April 18, 2021; Accepted: November 20, 2021.

Computer Science and Information Systems 19(2):639–658 https://doi.org/10.2298/CSIS210501053K

Transfer Learning and GRU-CRF Augmentation for Covid-19 Fake News Detection

Andrea Stevens Karnyoto, Chengjie Sun, Bingquan Liu, and Xiaolong Wang

School of Computer Science and Technology,
Harbin Institute of Technology,
Harbin 150001, China.
andre@ukitoraja.ac.id
cjsun@insun.hit.edu.cn
liubq@hit.edu.cn
wangxl@insun.hit.edu.cn

Abstract. The spread of fake news on online media is very dangerous and can lead to casualties, effects on psychology, character assassination, elections for political parties, and state chaos. Fake news that concerning Covid-19 massively spread during the pandemic. Detecting misinformation on the Internet is an essential and challenging task since humans face difficulty detecting fake news. We applied BERT and GPT2 as pre-trained using the BiGRU-Att-CapsuleNet model and BiGRU-CRF features augmentation to solve Fake News detection in Constraint @ AAAI2021 - COVID19 Fake News Detection in English Dataset. This research proved that our hybrid model with augmentation got better accuracy compared to our baseline model. It also showed that BERT gave a better result than GPT2 in all models; the highest accuracy we achieved for BERT is 0.9196, and GPT2 is 0.8986.

Keywords: Covid-19 fake news, hybrid neural network, Transfer Learning, Augmentation.

1. Introduction

In recent years, many phenomena have emerged and spread on the Internet, especially regarding the proliferation of information dissemination in online media. Some of the negative phenomena are hoaxes, rumors, and misinformation. The spread of fake news on online media is very dangerous [1,2]. And the effects can lead to casualties [3], psychological effect [4,5], character assassination [5], elections for political parties [6], and state chaos [7]. Fake news that concerning Covid-19 spread massively resulted in misunderstandings of information to the national and global communities during the pandemic. Detecting this misinformation on the Internet is an important and challenging task since even humans face difficulty in detecting fake news. In other words, humans cannot accurately distinguish whether it is fake or true news, especially it needs a tedious activity such as collecting evidence and sifting through facts. Therefore, our research concerns detecting fake news that related to covid-19 by using the Constraint @ AAAI2021 - COVID19 Fake News Detection in English Dataset [8] with Natural Language Processing Approaches Based.

The dataset for training and testing is provided by the "Constraint shared task" organizer [9], which aims to fight fake news related to COVID-19 across social media platforms such as Twitter, Facebook, Instagram, and other popular press releases. The dataset consists of 10,700 social media posts categorized into two labels: real and fake, all those written in English. Several previous studies have contributed to this Constraint @ AAAI2021 - COVID19 Fake News Detection in English shared task. Azhan et al. [10] apply a Layer Differentiated training procedure for training a pre-trained ULMFiT, Kakwani et al. [11] compile the IndicGLUE benchmark for language, Baris et al. [12] propose a modeling framework for those features by using BERT language model and external sources. Considering the number of researches utilizing the dataset, we think it crucial to contribute to this shared-task by using another approach.

Natural Language Understanding (NLU) is a branch of artificial intelligence (AI) that uses computer software to understand input presented in the text or speech format [13]. NLU is applied in automated reasoning, machine translation, question answering, news-gathering, text categorization, voice-activation, archiving, and large-scale content analysis [14,15,16]. We used Natural Language Understanding to do text categorization because it is more intelligent and efficient, which significantly challenges the semantic understanding in the system's module. We apply and modify the deep learning model conducted by Pin Ni et al. (2020) [17]: Natural Language Understanding approaches based on Intent Detection and Slot Filling joint tasks. They used the model BERT-RCNN-(BiGRU-CRF) and BERT-BiGRU-Att-CapsuleNet-(BiGRU-CRF), and they got pretty significant results. Unlike Pin Ni et al. (2020), we not only applied BERT but also employed GPT2 to training and testing our model for input pre-trained model. Also, we used BiGRU-CRF not for filling joint tasks but for feature augmentation.

Our contributions are as follows:

1) We performed two model structures: BiGRU-Att-CapsuleNet-(BiGRU-CRF). Also, we tested the dataset on a simple LSTM, Bi-GRU, BiGRU-Attention, and BiGRU-Attention-Capsule as a baseline for comparison toward our approach. Our hybrid model structure proves high competitiveness.
2) We used BERT and OpenAI GPT2 as pre-trained to all models.
3) We are involved in Constraint @ AAAI2021 - COVID19 Fake News Detection dataset shared task.

Additionally, as a study concerning hybrid-based (BiGRU(RNN), Attention(CNN), Augmentations(RNN)) and focusing on features augmentation methods, the hybrid neural network-based task model can improve model accuracies. It is proven our proposed model accuracy better compare to the baseline accuracy. The rest of the paper is organized in the following manner. In Section 2, we formally define related work in fake news detection. Section 3 describes our proposed method (the dataset, the main model, and the explanation of each layer). Section 4 is Experiment and Task. In Section 5, we present the Result and Analysis. Section 6 concludes the paper.

2. Related Work

Fake news detection is classified into Text Classification or Text Categorization. Some Fake News Detection studies use Machine Learning [18,19,20], and others use Deep

Learning [21,22]. Those techniques can also be generally categorized as News Content-based learning and Social Context-based learning. News content-based approaches deal with the different writing styles of published news articles, focusing on extracting several fake news articles related to both information and the writing style. Whereas Social context-based approaches deal with the latent information between the user and news article. The social engagements on articles can be a significant feature for fake news detection (to find the semantic relationship between news articles and writers) [23]. In the Fake News Detection research field, many datasets can be used, such as PolitiFact [24,25], Fake News Kaggle [18,26], The Fake News Challenge (FNC-1) [27,28], and Constraint@AAAI2021 - COVID19 Fake News Detection [9,10,12,11]. Ahmad et al. [18] developed Fake News Detection Using Machine Learning Ensemble Methods consists of Logistic Regression, Support Vector Machine, Random Forest (RF), etc. Monti et al. [21] proposed learning fake news-specific propagation patterns by exploiting deep geometric learning, a novel class of deep learning methods designed to work on graph-structured data. Konkobo et al. [24] performed a model to extract users' opinions expressed in comments. They used CredRank Algorithm to evaluate users' credibility and built a small network of users involved in spreading a piece of given news. Xu et al. [27] presented a new system, FaNDS, that detects fake news efficiently. The system is based on several concepts used in some previous works but a different context. There are two main concepts: An Inconsistency Graph and Energy Flow. Azhan et al. [10] proposed a Layer Differentiated training procedure for training a pre-trained ULMFiT model. They also used unique tokens to annotate specific parts of the tweets to improve language understanding and gain insights into the model, making the tweets more interpretable.

2.1. Fake News with BERT-Based

Deep neural network architectures for Transfer Learning Approaches have achieved substantial advances in a range of natural language processing (NLP) tasks recently [29]. Google published BERT as a sophisticated pre-training transfer learning model, and it is the most significant update as one of the NLP algorithms in recent years. Pre-training and fine-tuning are the two steps in the BERT framework [30]. The prominent model architecture is based on a multi-layer bidirectional Transformer encoder, which comes from the original implementation described and delivered in the tensor2tensor library [31]. Several studies that have been done by using BERT for Fake News Detection: Kaliyar et al. [23] proposed a BERT-based (Bidirectional Encoder Representations from Transformers) deep learning approach (FakeBERT). They combined different parallel blocks of the single-layer deep Convolutional Neural Network (CNN) having different kernel sizes and filters with the BERT. Gundapu et al. [32] used an ensemble of three transformer models (BERT, ALBERT, and XLNET) to detecting fake news. This model was trained and evaluated in the context of the ConstraintAI 2021 shared task "COVID19 Fake News Detection in English". Gupta et al. [33] presented a simple approach that uses BERT embeddings and a shallow neural network for classifying tweets using only text and discuss our findings and limitations of the method intext-based misinformation detection.

2.2. Fake News with OpenAI GPT2-Based

Recently, Pre-trained Generative Transformer-2(GPT-2) is a machine learning for text processing was developed by OpenAi. The capability of the GPT-2 is the ability to process up to 1024 tokens. Unlike other pre-trained models, GPT-2 technology can flow all tokens through the pre-training decoder generative block to provide good accuracy. Wang and Cho [34] declared that the GPT-2 generation gives good quality, powerful abilities, and minimal risk of error. On the other hand, the GPT-2 can generate text blocks such as short sentences that appear like written by humans, which means easy to generate fake text. Several studies solved fake news detection using OpenAI GPT-2: Harrag et al. [35] used GPT2-Small-Arabic generated fake Arabic Sentences. For evaluation, they compared different recurrent neural network (RNN) word embeddings-based baseline models, namely: LSTM, BI-LSTM, GRU, and BI-GRU, with a transformer-based model. Ranade et al. [44] generated fake CTI text descriptions using transformers automatically. They showed that given an initial prompt sentence, a public language model like GPT-2 with fine-tuning could generate plausible CTI text with the ability of corrupt cyber-defense systems.

2.3. Fake News with RNN-Based

A recurrent neural network (RNN) is an artificial neural network that uses sequential data or time-series data. It is powerful for modeling sequence data such as time series or natural language. The Advantages of RNN: Ability to process the input of any length, model size not increasing even it has different size of the input, all computation value is saved into account historical information, and the value of the weights are shared in all timeline. Singh et al. [36] proposed a framework that includes infrastructure to collect Twitter posts that spread false information. Their model implementation utilized the Transfer Learning scheme to transfer knowledge gained from a large and general fake news dataset to relatively more minor fake news events occurring during disasters as a means of overcoming the limited size of our training dataset. Ishiwatari et al. [37] proposed relational position encodings that provide Relational Graph Attention Networks (RGAT) with sequential information reflecting the relational graph structure. Accordingly, the RGAT model can capture both the speaker dependency and the sequential information.

2.4. Fake News with CNN-Based

CNN is a robust neural network with general-purpose functionalities in computer image and natural language processing; also, CNN can extract Euclidean structured data's spatial features. A small area sliding window is used in the CNN process to extract local features and then aggregates these features by pooling. The primary purpose of CNN is to reduce the number of parameters to a small extent but can effectively extract features over different matrix regions. Moreover, CNN plays a vital role in the information processing field. It employs 2D Convolution to process tokens of sentence embedding. It

is pretty simple because every input only has a 2-dimensional matrix of tokens, and the output is also a 2-dimensional matrix having a smaller size than the input. In the fake news detection task, several studies experimented with CNN: Goldani et al. [45] used Convolutional Neural Networks (CNN) with margin loss and different embedding models proposed for detecting fake news. They compared static word embeddings with the non-static embeddings that provide the possibility of incrementally up-training and updating word embedding in the training phase. Lu et al. [38] developed a novel neural network-based model Graph-aware Co-Attention Networks (GCAN) to achieve the goal. Extensive experiments conducted on real tweet datasets exhibit that GCAN can significantly outperform state-of-the-art methods. Mandelli et al. [39] proposed a 2-channel-based CNN that learns how to compare camera fingerprint and image noise. The proposed solution turns out to be much faster than the conventional approach and ensures increased accuracy. The method makes the system particularly suitable in scenarios where large databases of images are analyzed, like over social networks.

2.5. Fake News with Hybrid-based

Hybrid models are constructed of different neural networks (linear neural network and multi-layer neural network). The proposed hybrid system can be applied to many applications such as function approximation, time series prediction, and pattern classification, and text classification [40]. The hybrid model's output is formed of two or more different network yields. In one hybrid model, at least two types of neural network sets were considered together [41]. Nasir et al. [42] proposed a novel hybrid deep learning model that combines convolutional and recurrent neural networks for fake news classification. The model was successfully validated on two fake news datasets (ISO and FA-KES), achieving detection results that are significantly better than other non-hybrid baseline methods. Song et al. [43] proposed a multimodal fake news detection framework based on Crossmodal Attention Residual and Multichannel Convolutional Neural Networks (CARMN). The Crossmodal Attention Residual Network (CARN) can selectively extract the relevant information related to a target modality from another source modality while maintaining its unique information. In this research, we used the same model in Ni et al. [17]. Ni et al. use the hybrid model to solve the problem "Natural language understanding approaches based on the joint task of intent detection and slot filling for IoT voice interaction."

3. Proposed Method

3.1. Dataset Statistics

The Constraint @ AAAI2021 - COVID19 Fake News Detection in English Dataset [8] provided the shared task, which contains 10,700 humans annotated from media articles and posts acquired from multiple platforms. It is divided into data training (6,420 rows),

validation (2,140 rows), and test (2,140 rows). The unique words in the training dataset are 30,046, most length tokens are 1,481, and balance data distribution for Real and Fake labels. The dataset contains the post ID, tweet, and their corresponding label fields.

Table 1. Data Distribution for Constraint @ AAAI2021 - COVID19 Fake News Detection

Data	Real	Fake	Total	Unique Word
Train	3,360	3,060	6,420	30,046
Validation	1,120	1,120	2,140	13,697
Testing	1,120	1,120	2,140	14,121

Table 2. Some Post Fake and Real

Label	Post
Real	This #FourthOfJuly weekend if you choose to spend time outdoors at an event or gathering stay 6 ft apart & wear a cloth face cover to slow the spread of #COVID19. Learn more at https://t.co/c4F0aouMLd. https://t.co/u5tTl3m572
Real	We launched the #COVID19 Solidarity Response Fund which has so far mobilized $225+M from more than 563000 individuals companies & philanthropies. In addition we mobilized $1+ billion from Member States & other generous to support countries-@DrTedros https://t.co/xgPkPdvn0r
Fake	@realDonaldTrump has shifted his focus at different moments in the #CoronavirusOutbreak. We updated our running timeline of his response to the virus. https://t.co/pgXjssaRCB Reply to us with any recent Trump moments you think belong on this running list. https://t.co/g4WYcppDSO
Fake	RT @EllenCutch: Coronavirus misinformation is moving offline. A reddit user posted this flyer to the site and told us it had been delive…

Table 2 shows post tweets containing URL, Mention, Retweet, Hashtag, HTML special entities, and Number.

3.2. Data Preprocessing

First, we executed our tweet preprocessing and text preprocessing for transformer-based models by removing useless punctuation marks for text classification. We kept symbols '@' and'#' because those have specific function in tweets. Second, we changed the text into lowercase and replaced URLs, mentions, and emojis into particular tokens. Third, we utilized the Python emoji library to exchange the emoji with a short textual description: redheart:, :thumbsup:, etc. Furthermore, we transformed hashtags into words ("#DESEASE"→"DESEASE").

3.3. Main Model

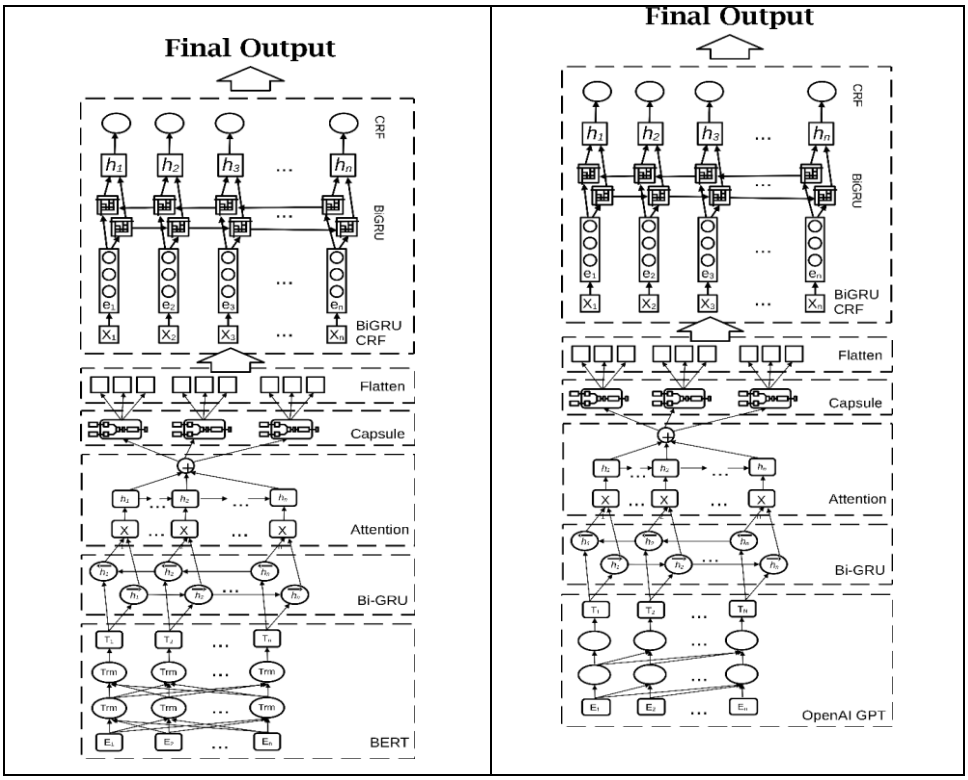


Fig 1. Main model

Fig 1. shows our two models that we were performed. The difference between the two models is the pre-training part before the first Bi-GRU layer was processed. The first model used BERT pre-trained, and the other used OpenAI GPT2 pre-trained.

Our model consists of four main parts:

1) Bidirectional GRU in the first layer, this layer receives the input from BERT.
2) The attention Layer is used to extract important features from the content.
3) The next layer is the capsule network layer, which is used to present each output of neurons with different intensity connections.
4) And the last layer is BiGRU-CRF; this layer is features segmentation that processes the capsule network features output.

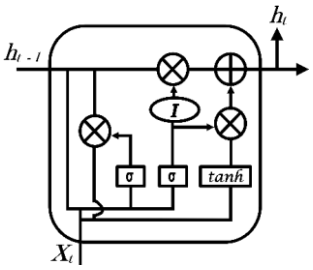


Fig 2. Gate Recurrent Unit

The first layer of our model is the Bidirectional Gate Recurrent Unit (BiGRU) layer. GRU is one of the LSTM variants with more advantages such as more superficial structural characteristics than standard LSTM, GRU has fewer parameters, and GRU has better convergence (Fig 2 shows GRU neuron structure). The two main parts of the GRU are the update gate and the reset gate. GRU uses the *z* Update Gate to manage the degree of impact at the previous time (*t - 1*) at the current hidden layer. The reset gate *r* is used as a control mechanism to ignore the output of the hidden layer information or not. The larger the update gate's value, the more the hidden layer's output is influenced by the previous layer. Moreover, smaller result value of the update gate means a lot of information was ignored at the last hidden layer. For more details, see the following formula:

$$
\begin{aligned}
r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\
z_t &= \sigma(W_i \cdot [h_{t-1}, x_t]) \\
\tilde{h}_t &= \tanh(W_c \cdot [r_t \cdot h_{t-1}, x_t]) \\
h_t &= (1 - z_t) \cdot c_{t-1} + z_t \cdot \tilde{h}_t
\end{aligned} \tag{1}
$$

We created a forward or backward GRU network model for the context in order to accomplish the bidirectional process of the text (from the beginning to end and vice versa). Both unidirectional GRUs together determines the output performance in the right and opposite directions, respectively. This layer is not only providing input information at each moment but also the bi-unidirectional GRUs jointly to determining the output for next moment (*t+1*). Layer that consists two unidirectional GRUs can considerably the bidirectional GRU, the weighted summation of the hidden layer state's output in the forward direction h_{t-1} and the hidden layer state in the backward direction (h_{t-1}): obtains the hidden state of the BiGRU at time *t*. See the following formula:

$$
\begin{aligned}
\overrightarrow{h_t} &= GRU(x_i, \overrightarrow{h_{t-1}}) \\
\overleftarrow{h_t} &= GRU(x_i, \overleftarrow{h_{t-1}}) \\
h_i &= w_t \overrightarrow{h_t} + v_t \overleftarrow{h_t} + b_t
\end{aligned} \tag{2}
$$

The input word vector from BERT/OpenAI GPT2 is transformed into nonlinear by using GRU. The length of input and output vector of GRU is different, the output size is

adjusted to the next layer input size. The hidden layer forward state processing w_t and v_t in current time generates h_t. Meanwhile, h_t is also used for GRU hidden layer backward state, which processing at time t. The hidden layer at present t also has a bias b_t. Next, to produce vector h_i for each word, the forward and backward GRU outputs are combined. Finally, each recurrent unit can process dependencies at different times.

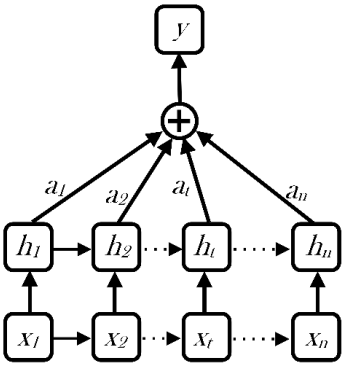


Fig 3. Attention Layer

The next layer is Attention as show in Fig 3. This mechanism was first proposed by the Google Mind Team. The main task of attention is to get important features from the bidirectional GRU layer output, and the motivation behind the incorporation of the attention make a network capable of learning object attentive features. In other words, attention mechanism focusses important information by simulate the attention characteristics of human brain. The output of each previous BiGRU layer is imported into the attention mechanism, and the result of this layer are specific array which will process in next layer. For instance, the terms "wonderful algorithm but the code was difficult", "wonderful" and "difficult" are all sentimentally inclined. It is sentiment polarity is more likely to be positive for the target "algorithm" because "wonderful" is closer to "algorithm".

$$
\begin{aligned}
s &= \sum_{i=1}^{l} \alpha_i h_i \\
\alpha i &= \frac{\exp(e_i)}{\sum_{j=1}^{n}(e_j)} \\
e_i &= v_i \tanh(w_i h_i + b_i)
\end{aligned}
\tag{3}
$$

The summation of the multiplication the coefficient α_1 and hidden layer state h_1 result from initial hidden layer state to updated hidden layer state in the initial input generates vector S. Weight coefficient matrix in *i-th* time is denoted by v_i and w_i. Corresponding offset at the *i-th* time denoted by b_i and e_i represents the value determined by the hidden layer state vector h_i at the *i-th* time.

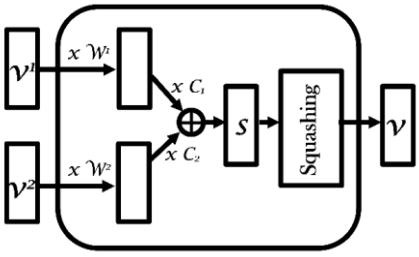


Fig 4. Capsule Network

Fig 4 shows the Capsule Network is a CNN model alternative with a slightly different operation than regular CNN, and it has a hierarchical relationship. In this study, dynamic routing aims to train neural networks to analyze the relationship between words in a vector and get characteristics in a text. The input of this layer is a vector generated from the previous attention layer. Simultaneously, the output is a vector consist of a probability of observation and a position for that observation. The following equation provides an overview of the process of the capsule network. The input capsule network is written with the V_i^t symbol.

$$\hat{u}_{j|i}^t = W_{ij}^t v_i^t$$
$$s_j^t = \sum_i c_{ij} \hat{u}_{j|i}^t$$
$$c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})} \tag{4}$$

To updated through the dynamic routing process, we are using coupling coefficient is denoted by C_{ij}. W_{ij}^t is a weight matrix that transforms input features from beginning to end. S_j^t is a global feature based on all input features. The coupling coefficients between global features T_j and all the input features sum into 1 and are determined by a "routing softmax" with B_{ij} initialized to 0. Then, to scale the globally represented modulus length between 0 and 1 uses the squash function:

$$g_j^t = squash(s_j^t) \tag{5}$$

Algorithm 1. shows how the dynamic routing process is executed.

```
procedure ROUTING(Uji, r)
 for all input feature i in input layer:
 for all global feature j in output layer: bij=0
   for r iterations do
     cij = softmax(bij)
     stj = Sigma(Cij x Utji)
     gtj = squash(stj)
     bij = bij + (Utji x Gtj)
```

Variable **r** is the number of iterations, and **Gtj** is one of the global features based on the input features. Therefore, for **j** = 1, …n_c, and all **U**tj in $\mathbf{U}^c$ or $\mathbf{U}^l$, we can generate two global representations, respectively.

BIGRU-CRF

Our approach utilizes a BiGRU architecture and CRF (Conditional Random Fields) for features segmentation. More precisely, this last layer involves three sub-layers: 1) an input layer containing flatten of capsule vectors, 2) a hidden layer where the Bi-GRU maps vectors to hidden sequences, and 3) an output layer that calculate the probability of label base of previous hidden sequences. The CRF model is a discriminant undirected graphical probability. It has been successfully applied in various natural language processing. Linear chain CRFs are most popular in NLP tasks, which implement sequential dependencies in the predictions and consist of undirected graph learning, based on the maximum entropy and hidden Markov, but simpler compare standard hidden Markov models.

$$p(\hat{y} \mid \hat{x}; w) = \frac{\exp(\sum_i \sum_j w_j f_j(y_i - 1, y_i, \hat{x}, i))}{\sum_{y_z \in y} \exp(\sum_i \sum_j w_j f_j(y_i - 1, y_i, \hat{x}, i))} \tag{6}$$

Formula 6 shows a form of CRF model. Let $x = \{x_1, x_2, \cdots, x_n\}$ denote the observation sequence and $y = \{y_1, y_2, \cdots, y_n\}$ be the set of finite states. W is the weight vector for weighing the output feature vector f from the BiGRU. The Viterbi algorithm will perform training and decoding.

4. Experiment and Task

4.1. Experiment Setup

The experiments run on Intel Core (TM) i7 8700, 6 core 3.20GHz Processor, 16 GB RAM, Nvidia GeForce GTX 1050 Ti GPU 4 GB. Base program and training use Python 3.7.8 and TensorFlow Version: 2.1.0. Pre-training and tokenizer use BERT transformer 3.4.0 with PyTorch 1.6.0+cu101. Several important hyper-parameters determine this architecture: Training model learning-rate= 0.001, epoch 10, batch-size 8. The dimension of word embedding for model inputs is different depend on datasets.

4.2. Description of Task

We divided tasks into two steps. 1) Prepare a pre-trained process using BERT and GPT2, 2) Train the Model using datasets. First, we took a maximum of 300 features

(words) using the NLTK tool in the preprocessing before entering the pre-training process. However, BERT and GPT2 tokenizers generate a different number of tokens in the pre-training. BERT generates 375 tokens while GPT2 generates 413 tokens from those 300 features, so the system automatically adjusted the maximum padding and input layer number referred to BERT and GPT2 output. Secondly, we conducted model training. We train for BERT on the BERT-LSTM, BERT-BiGRU, BERT-BiGRU-Attention, BERT-BiGRU, BERT-BiGRU-Attention-Capsule, BERT-BiGRU-Attention-Capsule-BiGRU-CRF models. Also, we were performing for OpenAI-GPT2 on the GPT2-LSTM, GPT2-BiGRU, GPT2-BiGRU-Attention, GPT2-BiGRU, GPT2-BiGRU-Attention-Capsule, GPT2-BiGRU-Attention-Capsule-BiGRU-CRF models. After completing all training processes towards models, we calculate and compare the testing results, such as accuracy, loss, F1, and the recall score.

5. Result and Analysis

To further prove that our proposed model can better accuracy by capturing more features details and enhancing the dependency between layers. We evaluated our proposed model (BiGRU-Attention-CapsNet-(BiGRU-CRF)) and baseline systems (LSTM, BiGRU, BiGRU-Attention, and BiGRU-Attention-CapsNet) toward The Constraint @ AAAI2021 - COVID19 Fake News Detection in English Dataset. Experiments applied the same hyperparameters such as fine-tuning, learning-rate, and batch-size settings. We present experimental results in Table 3 to prove that the techniques discussed in our proposed method contribute to increasing Neural-Network-based binary classification performance and then compare all models on the datasets mentioned above to get an overall impression of their performance. Recall, F1-Score and accuracy have been determined from the confusion matrix, and we used those results to decide classification results. The BERT-BASED section in Table 3 shows although the highest training accuracy is BERT-LSTM (baseline), our proposed method got the best accuracy for testing. It indicates no rigid relationship between training accuracy and testing accuracy. When the training accuracy is the highest, it does not mean it will get the highest accuracy result for testing. In contrast to GPT2-BASED, our approach got the highest accuracy for training, validation, and testing. Meanwhile, the BERT-BiGRU-Attention-CapsNET model got unsatisfied accuracy for training both for BERT and GPT2 Pre-training, although the accuracy for testing is still higher than LSTM. We assume that the shuffling process for features is still in capsules form and has not entered the augmentation process. Moreover, we concluded that BERT Pre-trained achieves better accuracy than GPT2 Pre-trained, although both use the same models (baseline and our proposed model). Comparing the two Pre-trained testing accuracies for our proposed model is 0.0208, and we concluded that BERT has significant enough, even on the F1 and Recall results.

The BERT-Based accuracy and GPT2-Based accuracy for all Baseline and Our Proposed model curves after training versus the number of epochs for the classification task based are shown in Fig 5. We can observe the curves that the model has learned well and does not have any significant result swing of accuracy at the end of the epoch.

Table 3. Train, Validation, and Test Result

	Train		**Validation**		**Testing**		
	Acc	**Loss**	**Acc**	**Loss**	**Acc**	**F1**	**Recall**
BERT BASED							
BERT-LSTM	**0.9937**	**0.0193**	0.9061	0.5393	0.9046	0.9045	0.9050
BERT-BiGRU	0.9827	0.0895	0.9192	0.2975	0.9117	**0.9116**	0.9122
BERT-BiGRU-Attention	0.9748	0.0640	0.9168	0.6256	0.9079	0.9076	0.9074
BERT-BiGRU-Attention-CapsNET	0.9221	0.1489	0.9084	0.2350	0.9061	0.9057	0.9054
BERT-BiGRU-Attention-CapsNET-(BiGRU-CRF)	0.9899	0.1011	**0.9238**	**0.2013**	**0.9196**	0.9113	**0.9193**
GPT2 BASED							
GPT2-LSTM	0.9123	0.2523	0.8860	0.2594	0.8841	0.8841	0.8800
GPT2-BiGRU	0.9165	0.2131	0.9192	**0.2411**	0.8766	0.8741	0.8718
GPT2-BiGRU-Attention	0.9226	0.2110	0.9168	0.2692	0.8855	0.8855	0.8878
GPT2-BiGRU-Attention-CapsNET	0.8876	**0.2089**	0.9084	0.2725	0.8846	0.8843	0.8841
GPT2-BiGRU-Attention-CapsNET-(BiGRU-CRF)	**0.9409**	0.2098	**0.9238**	0.2424	**0.8986**	**0.8924**	**0.8897**

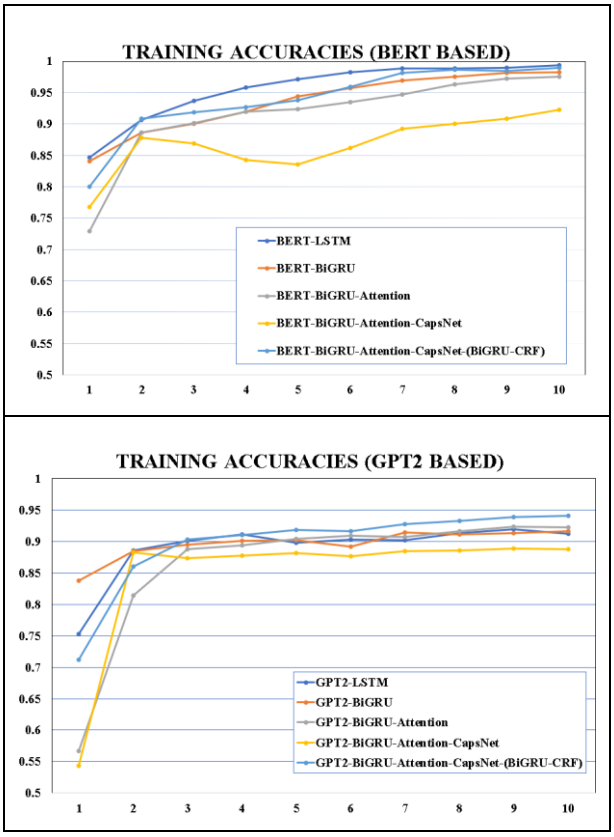


Fig 5. Training Accuracies

As shown in Fig 5, all models yield accuracies with only slight differences from one to another. It shows LSTM gets the best training results. The training system on LSTM (simple model) can only train non-hidden and distinctive features, leading to ease of learning for neural networks. BiGRU-Attention-CapsNet got the worst outcomes. We assume the model is not complete, and this neural network is still just starting to analyze the relationship between words and try to get the characteristics of the text. Although our approach (BiGRU-Attention-CapsNet-(BiGRU-CRF)) ranks number two for accuracy during training both for BERT and GPT2, this model obtained good accuracy during testing. However, the overall model provides increased accuracy in subsequent epochs.

5.1. Effect of Sentence Length

In our testing dataset, we found 1210 rows (56.54%) contain 15 words or less, 878 rows (41.02%) have 16-30 words, 47 rows (2.19%) contain 31-45 words and five rows (0.23%) contain more than 45.

Table 4. Effect of Sentences length toward classification accuracies

	Sentence Length			
	<=15	**16-30**	**31-45**	**>=46**
BERT-BASED				
BERT-LSTM	0.8726	**0.9568**	0.9352	0.6000
BERT-BiGRU	0.8769	**0.9636**	0.8723	0.6000
BERT-BiGRU-Attention	0.8727	**0.9567**	0.9362	0.6000
BERT-BiGRU-Attention-CapsNET	**0.9116**	0.8998	0.8936	0.8000
BERT-BiGRU-Attention-CapsNET-(BiGRU-CRF)	0.8810	**0.9761**	0.8723	0.8000
GPT2-BASED				
GPT2-LSTM	0.8791	**0.8942**	0.8733	0.6000
GPT2-BiGRU	0.8686	0.8884	**0.8936**	0.6000
GPT2-BiGRU-Attention	0.8760	**0.8998**	0.8723	0.8000
GPT2-BiGRU-Attention-CapsNET	0.8793	**0.8941**	0.8723	0.6000
GPT2-BiGRU-Attention-CapsNET-(BiGRU-CRF)	**0.9083**	0.8884	0.8723	0.6000

Table 4 shows the effect of sentence length from each model experiment. The sentences that contain 16-30 words got better accuracy for most models. Sentences that contain over 45 words have the lowest accuracy due to the uneven distribution of data. It indicates models we tested gain the best learning rates and predictions when the text contains 16-30 words in length. Our proposed model got the highest result in the BERT section, which is 0.9761. And the lowest is obtained by BERT-LSTM, BERT-BiGRU, BERT-BiGRU-Attention, which is 0.6. In the GPT2 section, our proposed model also gets the best result, which is 0.9083. In contrast to BERT, our proposed model gains the best results in the 1–15-word group for GPT2. But even so, the difference between one result to another is only slight. Because sentences longer than 45 are few in the dataset, the learning process in the network only has a few samples.

5.2. Most Common Terms in Fake News our Model can Detect

We also explored the most frequent words in our fake news dataset that our model can detect. However, we found the terms frequently used in fake news are similar to those that appeared most often in the entire dataset.

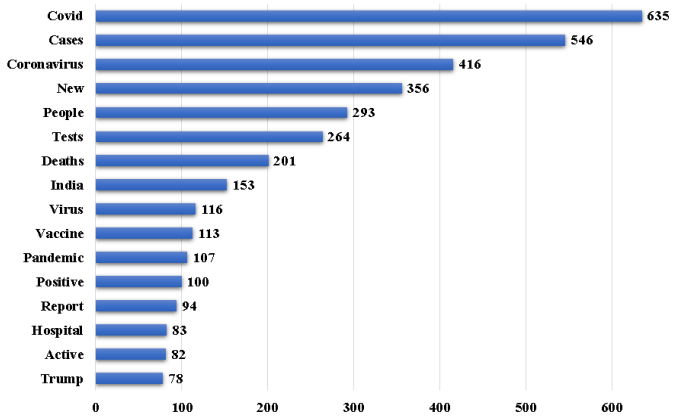


Fig 6. Most common term was detected by our models in The Constraint @ AAAI2021 - COVID19 Fake News Detection Dataset

Fig 6 shows the most frequent words in the testing dataset which our model can detect. The single text contains terms, which means every text in the dataset may consist of more than one most common word. Because the dataset for this research is related to covid-19, we only show the whole words related to covid-19. We noticed that "Covid," "Cases," and "Coronavirus" words are the most commonly appeared in this fake news dataset; however, those words are also common to appear in the news media and attract readers the most both for real or fake news. After the "India" term, the next frequently used words were not significantly different because some were described sufficiently in the top 3 words.

In the experiment, we train and test the performance of our proposed model and various baseline models. It showed the results of our proposed models are better than the baseline model. It also confirmed that our model has good observation and can catch complex augmentation and robust detection to improve the quality of the text classification.

6. Conclusion

With the growing popularity of online media such as online news, Facebook, Twitter, and other social media, more and more people get information from online media instead of newspapers and television. However, unresponsible people also used online media to spread fake news, and the effects make a negative impact on individual users and broader society. In this paper, we first exposed the interest and descriptions of automatic fake news detection. Then we compared and discuss our proposed (BiGRU-Att-CapsuleNet-(BiGRU-CRF)) model and our baseline (BiGRU, BiGRU-Attention, BiGRU-Attention-CapsuleNet). We also used BERT, GPT as pre-trained, and Constraint @ AAAI2021 - COVID19 Fake News Detection in English as a dataset to test our model and baseline. Based on our observations, our proposed method got better accuracies compared to baseline.

References

1. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2020). Fakenewsnet: A data repository with news content, social context, and spatiotemporal information for studying fake news on social media. Big Data, 8(3), 171-188.
2. Riedel, B., Augenstein, I., Spithourakis, G. P., & Riedel, S. (2017). A simple but tough-to-beat baseline for the Fake News Challenge stance detection task. arXiv preprint arXiv:1707.03264.
3. Apuke, O. D., & Omar, B. (2020). Fake news proliferation in Nigeria: Consequences, motivations, and prevention through awareness strategies. Humanities and Social Sciences Reviews, 8(2), 318-327.
4. Ozbay, F. A., & Alatas, B. (2020). Fake news detection within online social media using supervised artificial intelligence algorithms. Physica A: Statistical Mechanics and its Applications, 540, 123174.
5. Pulido, C. M., Ruiz-Eugenio, L., Redondo-Sama, G., & Villarejo-Carballido, B. (2020). A new application of social impact in social media for overcoming fake news in health. International journal of environmental research and public health, 17(7), 2430.
6. Maldonado, M. A. (2019). Understanding fake news: Technology, affects, and the politics of the untruth. Historia y Comunicación Social, 24(2), 533.
7. Waisbord, S. (2018). Truth is what happens to news: On journalism, fake news, and post-truth. Journalism studies, 19(13), 1866-1878.
8. Constraint-shared-task-2021, Available: https://constraint-shared-task-2021.github.io/ (current April 2021)
9. Akhtar, M. S., & Chakraborty, T. (2021). Overview of constraint 2021 shared tasks: Detecting english covid-19 fake news and hindi hostile posts. In Combating Online Hostile Posts in Regional Languages during Emergency Situation: First International Workshop, CONSTRAINT 2021, Collocated with AAAI 2021, Virtual Event, February 8, 2021, Revised Selected Papers (p. 42). Springer Nature.
10. Azhan, M., & Ahmad, M. (2021). LaDiff ULMFiT: A Layer Differentiated training approach for ULMFiT. arXiv preprint arXiv:2101.04965.
11. Kakwani, D., Kunchukuttan, A., Golla, S., Gokul, N. C., Bhattacharyya, A., Khapra, M. M., & Kumar, P. (2020, November). iNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings (pp. 4948-4961).
12. Baris, I., & Boukhers, Z. (2021). ECOL: Early Detection of COVID Lies Using Content, Prior Knowledge and Source Information. arXiv preprint arXiv:2101.05499.
13. Ovchinnikova, E. (2012). Integration of world knowledge for natural language understanding (Vol. 3). Springer Science & Business Media.
14. Van Harmelen, F., Lifschitz, V., & Porter, B. (Eds.). (2008). Handbook of knowledge representation. Elsevier.
15. Petrović, Đ., & Stanković, M. (2018). Use of linguistic forms mining in the link analysis of legal documents. Computer Science and Information Systems, 15(2), 369-392.
16. Zhao, H., Cao, J., Xu, M., & Lu, J. (2020). Variational neural decoder for abstractive text summarization. Computer Science and Information Systems, 17(2), 537-552.
17. Ni, P., Li, Y., Li, G., & Chang, V. (2020). Natural language understanding approaches based on joint task of intent detection and slot filling for IoT voice interaction. Neural Computing and Applications, 1-18.
18. Ahmad, I., Yousaf, M., Yousaf, S., & Ahmad, M. O. (2020). Fake News Detection Using Machine Learning Ensemble Methods. Complexity, 2020.

19. Gilda, S. (2017, December). Notice of Violation of IEEE Publication Principles: Evaluating machine learning algorithms for fake news detection. In 2017 IEEE 15th student conference on research and development (SCOReD) (pp. 110-115). IEEE.
20. Aphiwongsophon, S., & Chongstitvatana, P. (2018, July). Detecting fake news with machine learning method. In 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON) (pp. 528-531). IEEE.
21. Monti, F., Frasca, F., Eynard, D., Mannion, D., & Bronstein, M. M. (2019). Fake news detection on social media using geometric deep learning. arXiv preprint arXiv:1902.06673.
22. Sahoo, S. R., & Gupta, B. B. (2021). Multiple features based approach for automatic fake news detection on social networks using deep learning. Applied Soft Computing, 100, 106983.
23. Kaliyar, R. K., Goswami, A., & Narang, P. (2021). FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. Multimedia Tools and Applications, 1-24.
24. Konkobo, P. M., Zhang, R., Huang, S., Minoungou, T. T., Ouedraogo, J. A., & Li, L. (2020, November). A Deep Learning Model for Early Detection of Fake News on Social Media. In 2020 7th International Conference on Behavioural and Social Computing (BESC) (pp. 1-6). IEEE.
25. Oriola, O. Exploring N-gram, Word Embedding and Topic Models for Content-based Fake News Detection in FakeNewsNet Evaluation. International Journal of Computer Applications, 975, 8887.
26. Shakeel, D., & Jain, N. Fake news detection and fact verification using knowledge graphs and machine learning.
27. Xu, J., Zadorozhny, V., Zhang, D., & Grant, J. (2020). FaNDS: Fake News Detection System Using Energy Flow. arXiv preprint arXiv:2010.02097.
28. Hassan, F. M., & Lee, M. (2020, September). Multi-stage News-Stance Classification Based on Lexical and Neural Features. In Conference on Complex, Intelligent, and Software Intensive Systems (pp. 218-228). Springer, Cham.
29. Virtanen, A., Kanerva, J., Ilo, R., Luoma, J., Luotolahti, J., Salakoski, T., ... & Pyysalo, S. (2019). Multilingual is not enough: BERT for Finnish. arXiv preprint arXiv:1912.07076.
30. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
31. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. arXiv preprint arXiv:1706.03762.
32. Gundapu, S., & Mamid, R. (2021). Transformer based Automatic COVID-19 Fake News Detection System. arXiv preprint arXiv:2101.00180.
33. Gupta, A., Sukumaran, R., John, K., & Teki, S. (2021). Hostility Detection and Covid-19 Fake News Detection in Social Media. arXiv preprint arXiv:2101.05953.
34. Wang, A., & Cho, K. (2019). Bert has a mouth, and it must speak: Bert as a markov random field language model. arXiv preprint arXiv:1902.04094.
35. Harrag, F., Debbah, M., Darwish, K., & Abdelali, A. (2021). Bert transformer model for detecting Arabic GPT2 auto-generated tweets. arXiv preprint arXiv:2101.09345.
36. Singh, D., Shams, S., Kim, J., Park, S. J., & Yang, S. Fighting for Information Credibility: An End-to-End Framework to Identify Fake News during Natural Disasters.
37. Ishiwatari, T., Yasuda, Y., Miyazaki, T., & Goto, J. (2020, November). Relation-aware Graph Attention Networks with Relational Position Encodings for Emotion Recognition in Conversations. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 7360-7370).

38. Lu, Y. J., & Li, C. T. (2020). GCAN: Graph-aware co-attention networks for explainable fake news detection on social media. arXiv preprint arXiv:2004.11648.
39. Mandelli, S., Cozzolino, D., Bestagini, P., Verdoliva, L., & Tubaro, S. (2020). CNN-based fast source device identification. IEEE Signal Processing Letters, 27, 1285-1289.
40. Chen, Y., Kak, S., & Wang, L. (2008). Hybrid neural network architecture for on-line learning. arXiv preprint arXiv:0809.5087.
41. Rojek, I. (2010, June). Hybrid neural networks as prediction models. In International Conference on Artificial Intelligence and Soft Computing (pp. 88-95). Springer, Berlin, Heidelberg.
42. Nasir, J. A., Khan, O. S., & Varlamis, I. (2021). Fake news detection: A hybrid CNN-RNN based deep learning approach. International Journal of Information Management Data Insights, 1(1), 100007.
43. Song, C., Ning, N., Zhang, Y., & Wu, B. (2021). A multimodal fake news detection model based on crossmodal attention residual and multichannel convolutional neural networks. Information Processing & Management, 58(1), 102437.
44. Ranade, P., Piplai, A., Mittal, S., Joshi, A., & Finin, T. (2021). Generating Fake Cyber Threat Intelligence Using Transformer-Based Models. arXiv preprint arXiv:2102.04351.
45. Goldani, M. H., Safabakhsh, R., & Momtazi, S. (2021). Convolutional neural network with margin loss for fake news detection. Information Processing & Management, 58(1), 102418.

Andrea Stevens Karnyoto received the Master Engineering Degree in computer science from Universitas Hasanuddin, Makassar, Indonesia, in 2010. He is currently working toward the Ph.D. degree at the School of Computer Science, Harbin Institute of Technology, Harbin, China. His current research interests include Fake News Prevention and Detection, Text Mining, Natural Language Processing, and Artificial Intelligence.

Chengjie Sun received the Ph.D. degree from Harbin Institute of Technology, Harbin, China, where he iscurrently working on discriminative learning models for text mining. Since 2009, he has been an Associate Professor with Harbin Institute of Technology. His research interests include developing machine learning techniques for natural language processing and understanding.

Bingquan Liu received the Ph.D. degree in computer application technology from Harbin Institute of Technology, Harbin, China, in 2003. He is currently an Associate Professor with the School of Computer Science and Technology, and the Deputy Dean of Intelligent Technology and Natural Language Processing Research Group, Harbin Institute of Technology. His research interests include Question and Answering, Natural Language Processing, and Artificial Intelligence.

Xiaolong Wang received the Ph.D. degree in computer application technology from Harbin Institute of Technology, Harbin, China, in 1989. He is currentlya Professor with the School of Computer Science and Technology, Harbin Institute of Technology, China. He was honored as the outstanding contribution doctor in 1991, and the special allowance expert of the State Council in 1993. He was the inventor of the Chinese sentence level input method that embeddedin Microsoft Windows since 1996. His

research interests include intelligent input method, online finance information platform, question answering, and artificial intelligence.

Received: May 01, 2021; Accepted: September 30, 2021.

Computer Science and Information Systems 19(2):659–678 https://doi.org/10.2298/CSIS210507002W

Performance and Scalability Evaluation of a Permissioned Blockchain Based on the Hyperledger Fabric, Sawtooth and Iroha

Arnold Woznica and Michal Kedziora[1]

Wroclaw University of Science and Technology, Wroclaw, Poland
michal.kedziora@pwr.edu.pl

Abstract. This paper shows the performance and scalability evaluation of different blockchain platform implementations. Hyperledger Iroha implementing YAC consensus, Sawtooth implementing PoET algorithm, and Hyperledger Fabric framework implementation. Performance evaluation and scalability assessment were done by varying different sets of parameters such as block size, transaction sending rate, network traffic distribution, and network size. Performance evaluation was done based on average transaction latency, network throughput, and transaction failure rate. Scalability was assessed based on changes in transaction latency and throughput with increasing network size. Test results let to study the impact of a particular parameter on the private blockchain network performance and show how they can be adjusted to improve performance.

Keywords: Blockchain, Hyperledger Fabric, Sawtooth, Iroha.

1. Introduction

The popularity of blockchain technology paid attention to various businesses that want to adapt it to achieve their business goals. In a public blockchain network, everyone can join the network and be its participant without any authentication. To prevent fraud and malicious operations on the blockchain network, a particular blockchain platform must provide appropriate security mechanisms. In public blockchains, security is generally provided by the underlying consensus algorithms, such as proof of work in Bitcoin or proof of stake in Ethereum[12]. The consensus is the process by which a network of nodes provides a guaranteed ordering of transactions and validates the block of transactions [9]. High scalability is another issue that public consensus protocols must handle well. Unfortunately, those factors result in low throughput and high transaction latency in those systems, i.e, Bitcoin has throughput around 3 transactions per second and transaction latency around 10 minutes[14]. On the other hand, companies adapting blockchain technology work in a partially trusted environment, therefore the security of business blockchain networks might be resolved with mechanisms other than used consensus algorithms[11][10][3]. Scalability in business blockchain networks is also a less important factor. Business solutions need high throughput and low latency blockchain networks to handle quickly big amounts of transactions and that are the major demands on the consensus algorithms used. In 2015, companies, such as IBM and Intel, joined their efforts to create a common business blockchain framework under the Hyperledger project. Hyperledger is an open-source collaborative effort created to advance cross-industry blockchain

technologies. It is a global collaboration, hosted by The Linux Foundation, including leaders in finance, banking, Internet of Things, supply chains, manufacturing, and Technology [5]. It is a very fast-growing project concentrated on the development of a common framework for private and permissioned networks.

The goal and contribution of this work is evaluation of the performance of private blockchain platform implementations and algorithms. As there are many different consensus algorithms [16] the range of algorithms under evaluation is bounded to algorithms used in private and permissioned blockchain networks. The incentive behind such boundaries was a very fast growing Hyperledger project. Its growth means there is a very big demand on the market for private and permissioned blockchain solutions. The work aims to assess those consensus algorithms which are implemented in existing solutions. This work contains a performance evaluation of consensus protocols implemented in the following platforms: Hyperledger Fabric v1.4.0, Hyperledger Sawtooth v1.0.5, and Hyperledger Iroha v1.0.0_rc5. Performance Evaluation is performed by measuring: transaction latency, throughput, network, and scalability. It was done by testing with a different set of parameters such as transaction sending rate, network size, block size, and network traffic distribution.

2. Related work

This section describes articles that concentrate on the performance analysis of different consensus protocols of private blockchain platforms. Contributions of those papers were ideas on how to improve performance and they are implemented in newer versions of Hyperledger Fabric. There were not found any existing articles regarding performance analysis of Hyperledger, Iroha, and Sawtooth platforms, this makes a gap in the research field.

Nasir et al. [15] is the most related work to this paper. Authors concentrate on the performance evaluation of Hyperledger fabric v0.6 and fabric v1.0, those versions differ significantly, also other consensus protocols are used. Fabric v0.6 uses Practical Byzantine Fault Tolerance with order-execute architecture, on the other hand, fabric v1.0 uses Kafka ordering service with execute-order-validated architecture. To assess the performance, measurements such as transaction latency, execution time, throughput, and scalability are taken. The authors deployed the whole blockchain network on a single server machine with 24 core CPU and performed two tests: the first was an evaluation for a single peer network, the second was assessing scalability by varying the number of nodes to 20 in the network while performing the first test for each network topology. In the results section, the authors claimed that fabric v1.0 outperforms fabric v0.6. The second observation was that for fabric v1.0 the latency is within a certain range for different scenarios, while for fabric v0.6 the latency increases with network size.

Thakkar et al. [23] contains a very comprehensive performance evaluation of Hyperledger Fabric v1.0. The authors of that paper tried to answer the following questions: What should be the block size to achieve lower latency? What type of endorsement policy is more efficient? What is the performance difference between CouchDB and GoLevelDB when they are used as local world state databases? Authors divided transaction commit latency into endorsement, broadcast commitment, and ordering latency, which was measured by analyzing Hyperledger Fabric logs. The created test network was distributed

over different machines and consisted of four organizations with two endorsed peers each. For each experiment, many observations with guidelines on how to improve performance were given. Experiments included checking the impact of transaction arrival rate, block size, endorsement policy, ledger, database channels, and resource allocation. The main contributions of this paper were identifying three major performance bottlenecks, providing and studying improvements, changing the overall performance 16-times. All proposed improvements are contained in newer versions of the Hyperledger Fabric platform.

Sousa et al. [19] proposed a new Byzantine fault-tolerant Ordering Service for Hyperledger Fabric 1.0 instead of crash fault Kafka Ordering Service cluster. The proposed solution is based on BFT-SMaRt and WHEAT protocol. The authors implemented their solution and tested in LAN and WAN networks with nodes geolocated across the Americas.

Rüsch [18] mentions the performance and security issues of Byzantine fault tolerance schemes in blockchain and does an investigation for permissioned networks. The author refers to other papers to show the scalability problems of Byzantine fault-tolerant protocols for an increasing number of nodes.

Dinh et al. [6] presented the first Benchmark for private Blockchain evaluation called Blockbench. Blockbench can have an extended backend to evaluate different blockchain platforms. At the time of writing this article backend to test Ethereum, Parity, and Hyperledger fabric existed. Blockbench has a set of different macro and microbenchmarks to test different aspects of the working blockchain network. Blockbench supports evaluation of security by simulation network-level attacks. The authors compared three different platforms: Ethereum, Parity, and Hyperledger Fabric v0.6 by doing comprehensive tests. Empirical results have shown that Hyperledger Fabric v0.6 outperformed Ethereum and Parity across different benchmarks, but failed to scale over sixteen nodes. Their results have shown that tested platforms are not well suited to large-scale data processing workloads.

Li et al. [13] proposes a new architecture to improve blockchain network scalability. New architecture proposes satellite chains to create a network of networks, and it mentions the problem of scalability of single Byzantine Fault Tolerant based networks like Hyperledger Fabric v0.6. The presented architecture provides the ability to transfer assets between different networks in a secure way.

Gorenflo et al. [8] propose performance improvements on Hyperledger Fabric v1.2 without the change of its API. Authors test their implemented ideas to show that this platform can have a throughput of 20 000 transactions per second. Improvements consisted of I/O operations, caching, parallelism, and efficient data access. In their design, orderers receive only transaction IDs instead of full transactions, and validation on peers is more parallelized. Authors determined critical paths and leveraged light-weight data structures for fast data access.

Sukhwani et al. [20] [22] [21] presented a Stochastic Reward Net performance model of Hyperledger Fabric v1.0. This model can be used to compute the throughput, utilization, and mean queue length for each peer in a network. The model is validated with the usage of Hyperledger Caliper. Authors discovered the performance bottlenecks of the ordering service and ledger write operations. Authors claimed that the bottleneck can be mitigated using larger block sizes, although with transaction latency increase.

Turki et al. [24] proposed another Ordering service for Hyperledger fabric v1.0+ built on top of the HoneyBadgerBFT protocol. The authors tested their solution on a local machine and compared it to the results of Solo Ordering Service available for the Hyperledger Fabric v1+ platform.

Feng et al. [7] proposed a new Byzantine Fault Tolerant consensus called scalable dynamic multiagent PBFT (SDMA-PBFT), which is a modification of Practical Byzantine Fault Tolerance protocol. The proposed method decreases latency and improves efficiency and throughput.

Angelis et al. [4] compared the Practical Byzantine Fault Tolerance algorithm with proof of authority algorithms(Aura and Clique). Performance analysis was qualitative and based on how the algorithms worked in terms of message exchanging.

Vukolic [25] performed a theoretical comparison of Byzantine fault-tolerant algorithms versus Proof of work consensus algorithms with a literature review describing what was done to improve their performance and scalability.

Bakaman [1] made a comprehensive review of the state-of-the-art blockchain consensus algorithms. Then proposed an analytical framework to evaluate the pros and cons of consensus mechanisms.

Rui Wang [26] analyzed four mainstream blockchain systems (Ethereum, Fabric, Sawtooth and Fisco-Bcos), and then performed a performance comparison through open source blockchain benchmarking tools. After that, they propose optimization methods and discuss the future development of blockchain technique.

3. Methodology

Each of the evaluated platforms implements a different consensus mechanism. To test the performance of consensus, there is a need to submit transactions, which change the World State of a particular blockchain network. It is important to notice, that the database operation needed to trigger consensus mechanisms has an impact on the performance of a blockchain network. To reduce this impact simple transactions were needed to be implemented. This 'Simple' test works in the following way: Every transaction generates some random number based on the system time, which is appended to a literal value. Each of those literals creates a key-value pair with a constant value. Every transaction is an insert transaction causing World State change. The randomness of the key makes with negligible probability impossible to insert a key that was previously in the database, therefore transactions are protected against being invalid.

3.1. Research Environment

Experiments were conducted in the cloud on the Azure platform. A single virtual machine D64 v3 was used, with 64 vCPU, 256GB of RAM, and 1600GB storage. The running operating system was Ubuntu 18.04 LTS.

To analyze the performance of different Hyperledger platforms, a dedicated benchmark was used. Hyperledger Caliper [2] is a performance benchmark tool for multiple blockchain platforms that can cooperate with blockchain networks in two ways. First is connecting to an existing and working blockchain network, the second is starting a

blockchain network before the benchmark test using the networks defined in docker-compose.yaml files. Hyperledger Caliper produces .html reports containing several performance indicators, such as transactions per second, transaction latency, throughput, and resource utilization. Hyperledger Caliper architecture consists of four main layers: the benchmark layer, adapter layer, an interface layer, and blockchain framework layer. The adaptation layer is the main component of the caliper tool. It is used to integrate different blockchain network implementations into the benchmark. For every blockchain network, the adaptor implements the Caliper Blockchain NBIs(North Bound Interface) by using the corresponding blockchain's native SDK or RESTful API. Currently supported platforms are: Hyperledger Fabric v1.4+, Hyperledger Sawtooth 1.0+, Hyperledger Iroha 1.0 beta-3, and Hyperledger Burrow 1.0.

Benchmark layer contains the tests for typical scenarios, each test has a configuration file that defines test arguments and a backend blockchain network. Such configuration files define use cases that are then used by the underlying benchmark engine to perform the test. The role of this layer is to perform stress tests on the implemented blockchain platform. The blockchain framework layer contains different implementations of blockchain networks, which can be run separately from the benchmark. Caliper communicates with a particular network with the utilization of an appropriate adaptor.

3.2. Key Metrics Definitions

The first key metric is transaction latency, which is the time difference between transaction submission and transaction commit confirmation time across the network. It includes the propagation time and any setting time due to the consensus mechanism in place. This measure is computed per single transaction. The calculation of transaction latency differs according to the used protocol. This definition is appropriate for blockchain systems with deterministic transaction finality. Blockchain systems with a lottery-based consensus like Bitcoin, have probabilistic transaction finality, hence the latency calculation should follow other rules [17]. For measurements in this paper, the finality of the underlying systems under test is assumed. Second metric is transaction throughput, which is the rate at which the blockchain system under test commits valid transactions in the defined time frame. It is calculated as the number of transactions per second (tps). This rate refers to the entire blockchain network, rather than to a single node. Since only valid transactions are considered, the throughput consists of only positively verified transactions [21]. The transaction here refers to a commit transaction. This paper does not consider the throughput of query operations. Third metric is scalability, which is measured as the change in throughput and latency when increasing the number of nodes and the number of concurrent workloads. [6]

3.3. Test plan

There were four parameters for each blockchain platform test case: Transaction sending rate, block size, network size, and network traffic mode. A transaction sending rate is a number of transactions sent to the blockchain network in a defined time frame range. The transaction sending rate is presented as a transaction number per second. One test case consists of multiple rounds. In each round there is a constant transaction sending rate

being sent within 5 seconds. For example, for 50 Tps the total number of transactions sent in the round is 250 transactions. Each new round has a bigger transaction sending rate.

Block size is the maximum allowed transaction quantity which can be placed in a block being published to the blockchain network. Hyperledger Fabric and Hyperledger Iroha were tested with two different values of the block size(10 and 50). Hyperledger Sawtooth was tested only for a block size equal to 50.

Network size is the number of nodes participating in the blockchain network consensus mechanism. The smallest tested network consists of 5 nodes. The biggest tested network is for Hyperledger fabric and consists of 100 nodes.

The last parameter is the network traffic mode. Transactions can be distributed across all nodes in the network or can be sent to some particular nodes only. This parameter is introduced to check whether blockchain network performance will change when network traffic switches from distributed against all nodes to single node operating all transactions.

4. Results and discussion

This part contains the results of the tests for all parameters defined in the Test plan section. Test results are being shown in the form of figures and tables and the following examinations are being made:

- Impact of transaction sending rate to network latency.
- Impact of transaction sending rate to network throughput.
- Impact of block size to network latency.
- Impact of Block size to network throughput.
- Impact of network traffic mode to latency and throughput.
- Scalability, the impact of network size on latency.
- Scalability, the impact of network size on throughput.
- What are the reasons for failing transactions for some networks?

4.1. Hyperledger Fabric performance evaluation

This section contains the test results for Hyperledger Fabric v1.4. Fabric network consists of one organization owning all peers working on a single channel, therefore the network works as a private blockchain network. The network uses only one Solo ordering node. Endorsing peers are those who have smart contracts installed to be able to execute and endorse transactions. Due to technical reasons with running fabric network with Hyperledger Caliper maximum number of endorsing peers in all tests is 30, for network size exceeding 30 rest of peers works only as storage nodes. To analyze the impact of transaction sending rate on transaction latency, each fabric network was tested against different transaction sending rates. The whole network used a maximal block size of 10 transactions. Network traffic was distributed equally between all peers.

Figure 1 shows how the minimal transaction latency depends on transaction sending rates. To illustrate this relation three tested network sizes were chosen: 10, 50 and 100 peers. It is easy to notice that the network with 10 peers has smaller minimal transaction latency, but generally it is similar for all networks. Other observation is that the transaction sending rate does not have any impact on the minimal latency. For each fabric network size minimal transaction latency was bigger than 200ms.

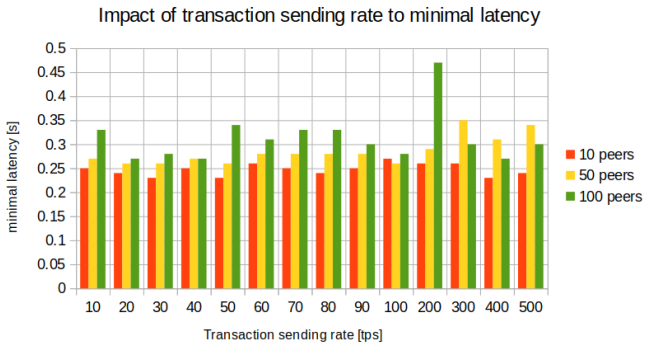


Fig. 1. Impact of Transaction sending rate to minimal latency for Hyperledger fabric platform

Figure 2 shows impact of transaction sending rate to average transaction latency in the network. The first observation is increasing the average transaction latency with increasing sending rate, it seems that at some point it grows faster, but this is because the scale on the x-axis has changed. It seems that the average latency depends linearly on the transaction sending rate. The second observation is that networks with fewer peers have smaller average latency than bigger networks at the same transaction sending rate. The third observation is that for networks with 30 peers and more there is no difference in average transaction latency. It might create an assumption that the performance of fabric networks depends only on the number of endorsed peers.

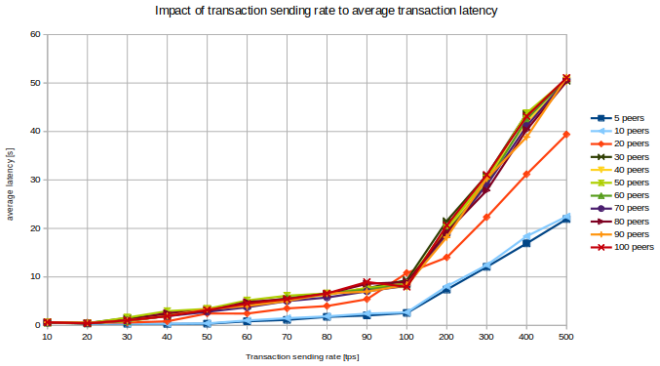


Fig. 2. Impact of transaction sending rate to average latency of Hyperledger fabric platform

Analysis of transaction sending rate impact to network throughput was performed just as its impact on latency. The maximum number of transactions in a single block is 10. Network traffic is distributed equally to all peers. The maximum number of endorsing peers in the network is 30.

The result shows how the network throughput depends on the transaction sending rate. Results show increasing network throughput with transaction sending rate to some point at which with larger transaction sending rate throughput is constant. To this point throughput is only a bit smaller than the transaction sending rate. The second observation is that smaller networks reach their saturation point later and achieve bigger network latency. As the maximum network endorsing peers number is limited to 30, we can see larger networks with no more endorsing peers have maximal throughput the same as the network with 30 peers only.

To analyze how block size impacts average latency fabric networks up to 100 peers were tested. Transactions were distributed equally between all peers. Tests were performed for two block size values: 10 and 50.

Figure 3 shows impact of block size to transaction average latency. Every network was tested against two-block size values. To detect the relations between maximal transactions in a block and transaction latency, network with 5, 20, and 40 peers were chosen to be depicted in the figure. Figure 3 shows there is no difference in transaction latency for different block sizes. It might suggest that the bottleneck of the transaction processing is not in the ordering service.

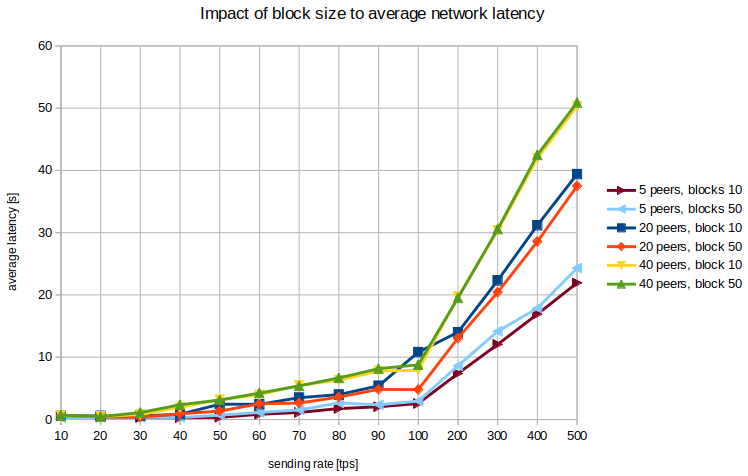


Fig. 3. Impact of block size to transaction latency for Hyperledger fabric platform

To analyze how block size impacts network throughput, fabric networks up to 100 peers were tested with a sending rate up to 500 transactions per second. Transactions were distributed equally between all peers. Tests were performed for two block size values: 10 and 50.

Results shows the impact of block size to network throughput. Every network was tested against two-block size values. To detect relations between network throughput and maximal transactions in a block fabric network with 5, 10, 20, and 40 peers were chosen to be depicted in the figure. Results shows that networks with 20 peers had bigger throughput

with larger block sizes while for 5 and 10 peers it was opposite, smaller block size caused better throughput. No strict correlation between block size and throughput were noticed.

To analyze the difference between traffic distribution modes to transaction latency and network throughput all network sizes were tested against with the transaction sending rate starting with 50 transactions per second and finishing with 500 transactions per second. The network block size was 50 for all networks. All transactions in the network are being handled by a single peer which is the only endorsing peer in the network.

Figure 4 shows the relation between transaction sending rate and average transaction latency for fabric networks up to 100 nodes with all network traffic sent to a single node. Average latency, as seen in the figure, is constant until a saturation point for a sending rate equal to 100 tps, after that the saturation point latency starts to grow linearly. In comparison with figure 2 from subsection "Analysis of transaction sending rate to transaction latency" it is easy to notice that for traffic divided between multiple nodes the latency was bigger.

Figure 5 shows the relation between transaction sending rate and network throughput for networks up to 100 nodes in which all transactions are sent to one peer. Network throughput, as seen in the figure, grows linearly with the transaction sending rate until it reaches a saturation point for the transaction sending rate between 100 and 200 tps. Regardless of the network size, all networks do not exceed the throughput of 150 tps in this examination.

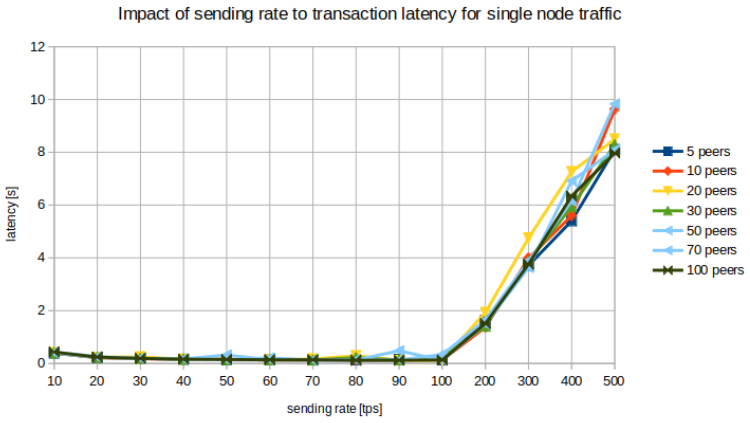


Fig. 4. Impact of transaction sending rate to transaction latency of Hyperledger Fabric platform for traffic handled by a single node

To analyze the relationship between network size and network average transaction latency fabric networks with peers quantity up to one hundred were tested. Each network was tested against a few sending transaction rates. All transactions sent to the network were distributed equally among all peers in the network. Maximal transactions in the block were equal to 10. The maximal number of endorsing peers was 30.

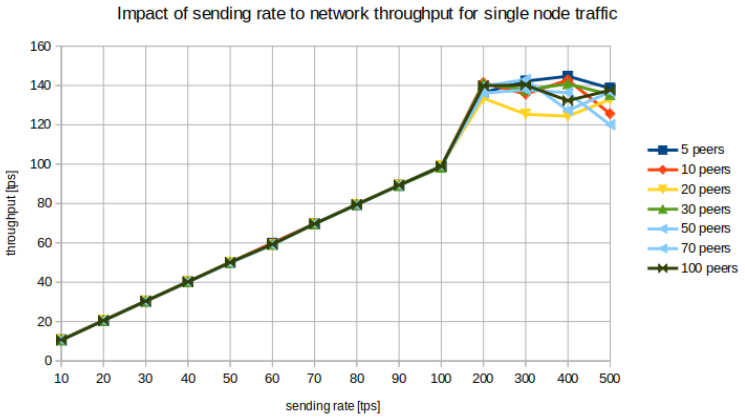


Fig. 5. Impact of transaction sending rate to network throughput for Hyperledger Fabric platform for traffic handled by a single node

Figure 6 shows impact of fabric network size to average transaction latency in the network. To show the relation between network size and its average transaction latency results are shown for different transaction sending rates: 50, 100, 300, and 500 transactions per second. From figure 6 it is easy to notice that the transaction latency grows with the network size up to the maximum number of endorsing peers in the network, which is 30. For more peers than 30 latency becomes a constant value, depending on sending transaction rate.

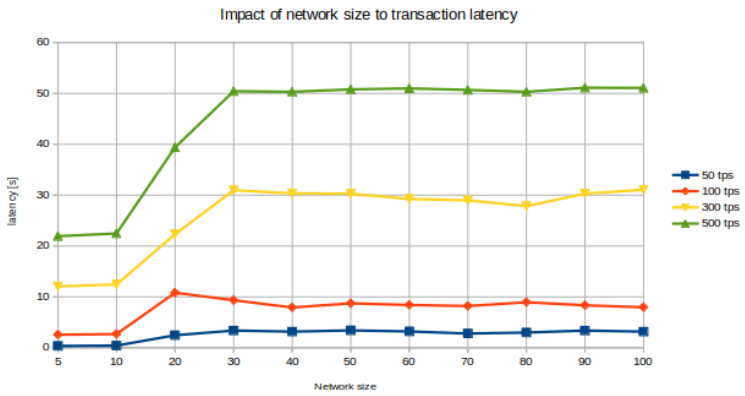


Fig. 6. Impact of network size to average transaction latency for Hyperledger Fabric

To analyze the relationship between network size and network throughput fabric networks with peers quantity up to one hundred were tested. Each network test consisted of sending transactions at a different rate. All transactions sent to the network were dis-

tributed evenly between all network nodes. Maximal transactions in the block were equal to 10. A maximal number of endorsing peers was 30.

Figure 7 shows impact of fabric network size to fabric network throughput. To show the relation between network size and network throughput each network was tested against different transaction sending rates: 50, 100, 300, and 500 transactions per second. Figure 7 shows that for bigger fabric network throughput is getting smaller. Peers who are not endorsing peers do not impact network throughput.

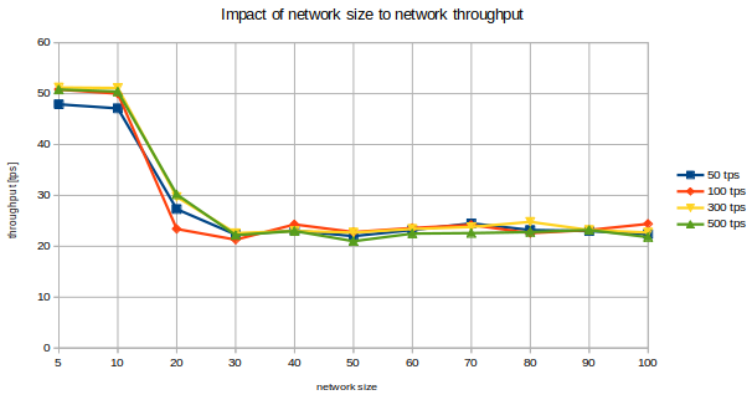


Fig. 7. Impact of network size to network throughput for Hyperledger Fabric

4.2. Hyperledger Iroha performance evaluation

This section contains test results and analysis of Hyperledger Iroha. This network was tested up to 50 nodes, as for more nodes the network did not work correctly. The network was tested for two maximal transactions in a single block value: 10 and 50. Maximal sending rate for which this platform was tested was 200 transactions per second.

Analysis of transaction sending rate to transaction latency To analyze the transaction sending rate impact to transaction latency each Iroha blockchain network was tested against different transaction sending rates. The whole network used a maximal block size of 50 transactions. Network traffic was distributed equally between all peers.

Figure 8 shows minimal transaction latency for different transaction sending rates. To illustrate these four network sizes were tested: 5, 10, 20, and 30 peers. Block size was equal to 50 transactions. Figure 8 shows that there is no strict correlation with transaction sending rate and minimal latency. For most transaction sending rates the minimal latency is around 4 seconds.

Results shows the average transaction latency for different transaction sending rates. To illustrate these five network sizes were tested consisting of 5, 10, 20, 30, and 40 peers. Block size was equal to 50 transactions. Result shows that generally the average transaction latency is constant and at the same value for all transaction sending rates, therefore

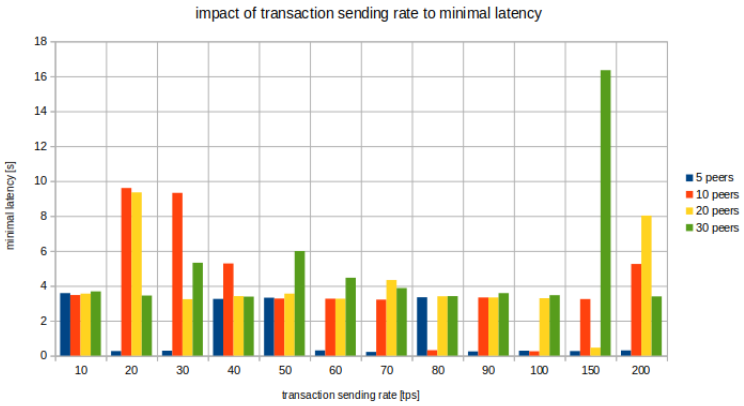


Fig. 8. Impact of transaction sending rate to minimal transaction latency for Hyperledger Iroha

does not depend on it. Network size seems to affect the value of average transaction latency, with larger networks having bigger average transaction latency.

Analysis of transaction sending rate impact to network throughput was preceded with testing Iroha networks up to 50 nodes against increasing the transaction rate up to 200 transactions per second. All networks used block size of 50 transactions and all sent transactions were distributed between all nodes in the network. Results shows the impact of sending transaction rate to Iroha network throughput for networks consisting of 5, 10, 20, 30, and 40 peers. The network of 40 peers has a constant throughput below 5 transactions per second for every transaction rate, but as seen in the previous section such a large network does not work correctly and most of the transactions fail. For smaller networks generally, the throughput is increasing with transaction sending rate, but all networks had throughput below 40 transactions per second, which is less than a maximal block size.

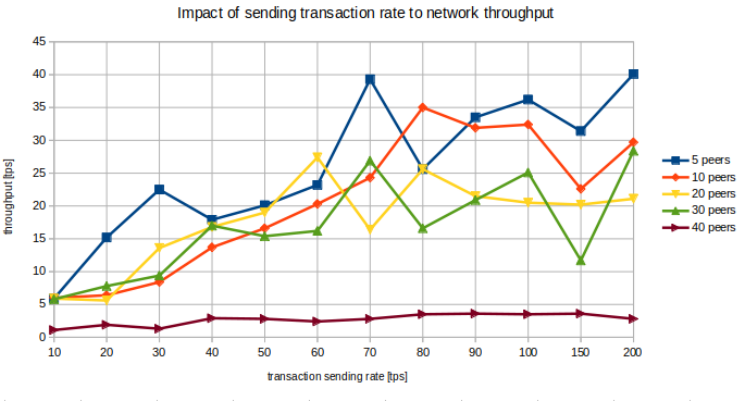


Fig. 9. Impact of transaction sending rate to network throughput for Hyperledger Iroha

Analysis of block size impact to average transaction latency was done for two block sizes: 10 and 50 transactions. The test was performed for three different network sizes: 5, 10, and 20 nodes. Each of the networks was tested against increasing the transaction sending rate up to 200 transactions per second. Transactions were distributed equally to all nodes in the networks.

Figure 10 shows impact of block size to average transaction latency in Iroha networks. To find the relation between block size and average transaction latency each network size had tested for different value of block size. After comparing appropriate pairs it is easy to see that for most transaction sending rates average latency was significantly lower for larger block size. Network with 20 nodes and block size equal to 50 has an average latency below 5 seconds for most transaction sending rates, while the same network with 10 transactions in a block can have an average latency around 43 seconds. This is 9 times bigger latency for 5 times smaller block.

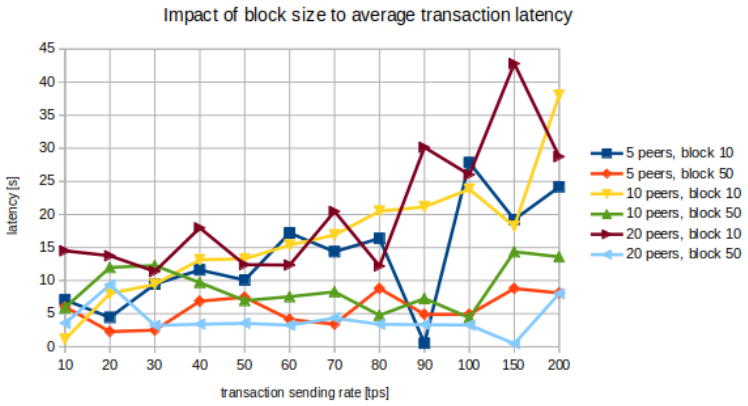


Fig. 10. Impact of block size to average transaction latency for Hyperledger Iroha

To analyze the impact of block size on network throughput tests with 10 and 50 maximal number of transactions in a block were performed. The test was performed for three different network sizes: 5, 10, and 20 nodes. Every network was tested against increasing the transaction sending rate up to 200 transactions per second. Transactions were distributed equally to all nodes in the networks.

Figure 11 shows block size impact to network throughput in Iroha networks. To find the relations between block size and network throughput networks must be compared in pairs: The same network sizes with different block sizes. For all networks depicted in figure 11 it is easy to notice that smaller block size causes the network throughput decrease. It is noticeable especially for bigger transaction sending rates.

This subsection describes the difference in the average transaction latency of Iroha networks for two cases:

- Network traffic distributed equally. Transactions are sent to every node in a network.
- Network traffic distributed to a single node that processes all transactions sent to the network.

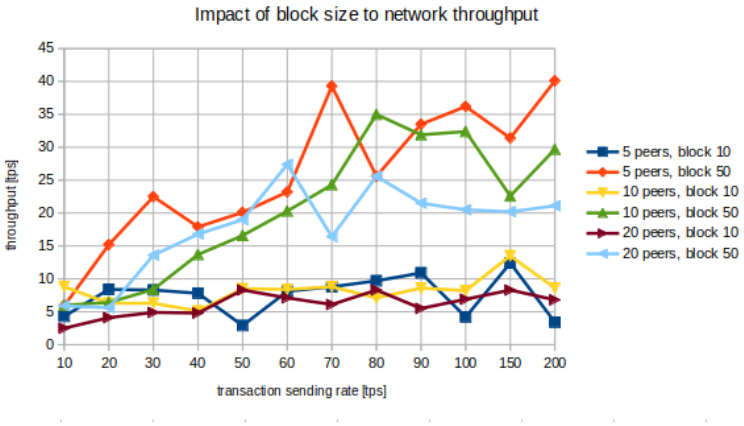


Fig. 11. Impact of block size to network throughput for Hyperledger Iroha

Figure 12 shows the impact of network traffic distribution to Iroha average transaction latency. All tests were performed by increasing transaction sending rate up to 200 transactions per second. The maximal transaction number in a block was 50. Each network size is tested with transactions sent to one node and distributed among all nodes in the network. For a network with 20 nodes traffic directing to a single node increases the average transaction latency for all transaction sending rates. For smaller networks such behavior is similar, but noticeable for higher transaction sending rates.

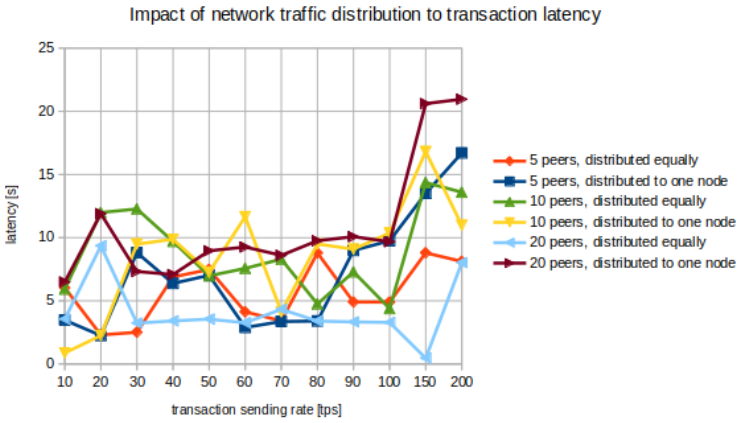


Fig. 12. Impact of network traffic distribution to average transaction latency for Hyperledger Iroha

Analysis of network traffic distribution to network throughput This subsection describes the difference in network throughput of Iroha networks for two cases:

– Network traffic distributed equally. Transactions are sent to every node in a network.
– Network traffic is distributed to a single node that processes all transactions sent to the network.

Figure 13 shows the impact of network traffic distribution on the Iroha network throughput. All tests were performed by increasing the transaction sending rate up to 200 transactions per second. Maximal block size is 50. Each network size is tested with transactions sent to one node and distributed among all nodes in the network. In figure 13 there is no strict correlation between the impact of transaction distribution to network throughput, but for transaction sending rate over 80 tps the throughput is higher for equal distribution of transactions between peers.

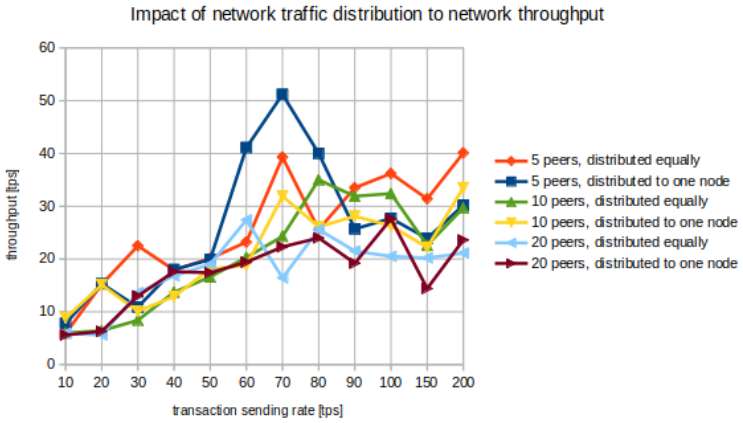


Fig. 13. Impact of network traffic distribution to network throughput for Hyperledger Iroha

Analysis of network size to transaction latency To analyze the relationship between network size and network average transaction latency of Iroha networks, networks with nodes number up to forty were tested. Each network was tested for increasing sending transaction rates up to 200 transactions per second. All transactions sent to the network were distributed equally against all peers in the network. Maximal transaction quantity in the block was equal to 50.

Figure 14 shows the impact of Iroha network size to the average transaction latency in the network. To show the relation between network size and its average transaction latency, results are shown for different transaction sending rates: 20, 60, 100 transactions per second. Figure 14 shows that average latency increases after network size exceeds 30 nodes. Between 10 and 30 nodes average transaction latency is generally constant and below 10 seconds. For a small Iroha network consisting of 5 nodes average transaction latency is the smallest.

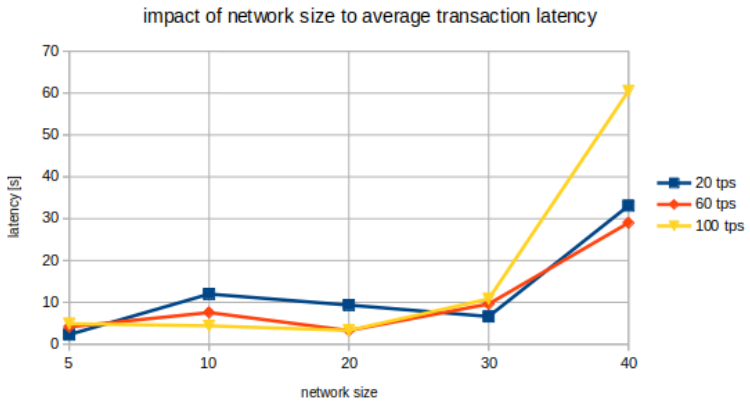


Fig. 14. Impact of network size to average transaction latency for Hyperledger Iroha

Analysis of network size to transaction throughput To analyze the relationship between network size and network throughput of Iroha networks, networks with nodes quantity up to forty were tested. Each network was tested for increasing sending transaction rates up to 200 transactions per second. All transactions sent to the network were distributed equally against all peers in the network. Maximal transaction quantity in the block was equal to 50.

Figure 15 shows that networks under small transactions sending rate(20 tps) have generally constant throughput when their size is between ten and thirty nodes. For high transaction sending rate(100 tps) throughput gets lower with network size increase. For network size exceeding 30 nodes throughput is very small(around 2-3 tps) regardless of transaction sending rate.

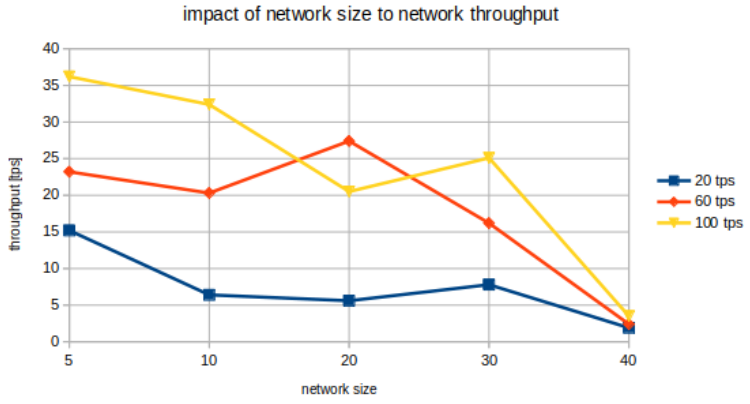


Fig. 15. Impact of network size to network throughput for Hyperledger Iroha

4.3. Hyperledger Sawtooth performance evaluation

This section contains test results and analysis of Hyperledger Sawtooth. Table 1 contains the parameters values for which benchmarking was done. This network was tested up to 30 nodes, as for more nodes network did not work at all. The network was tested for a block consisting of at most 50 transactions. The platform was tested for transaction sending rate increasing with every round starting with 10 and finishing with 100 transactions per second.

Table 1. Sawtooth performance evaluation results for network of 5 nodes and network traffic distributed among all nodes

sending rate[tps]	max latency[s]	min latency[s]	avg latency[s]	throughput[tps]	Failed [%]
10	4.81	0.41	2.5	7.3	0
20	5.22	0.42	2.99	13.9	0
30	2.82	0.41	1.74	24	0
40	2.81	0.42	2.34	27.4	0
50	6.23	1.61	3.91	26.5	0
60	8.26	0.61	4.9	25.1	0
70	9.02	0.62	7.37	2	67
80	NaN	NaN	NaN	NaN	100
90	6.23	4.62	5.35	1.4	67
100	8.02	5.42	6.7	2.4	49

Result shows that even small Sawtooth networks consisting of 5 peers only have a problem with successful transaction processing when network traffic is distributed among all participants. Networks with 20 nodes and more fail to process almost all transactions when the transaction sending rate exceeds 40 transactions per second. Similar results are in the scenario with a single node handling all transactions sent to network small networks (5 and 10 nodes) process almost all transactions successfully. For larger networks, transaction failures happens for smaller transaction sending rate. For example with a network consisting of 20 nodes shows that after rounds with 100% transaction failures(for 80 and 90 transactions per second) there is still a chance that in the next round network will process all transactions correctly.

There are multiple possible explanations why benchmarking of Hyperledger Sawtooth platform results in so many transactions failing including benchmark and platform problems:

- Hyperledger Caliper lets defining multiple nodes to which transactions are being sent, but only one validator which is used to check whether the transaction is added to the ledger. Due to the probabilistic implications of the PoET consensus algorithm, transactions can be included in other nodes, which are not yet in sync with the defined validator, therefore the transactions fail after some time.
- It was observed that right after sending transactions to a ledger there was an error message while checking the transaction state. There might be a bug in the Sawtooth adapter of the Hyperledger Caliper project. In the logs produced by peers, the network seemed to be in sync(as no attempts to synchronize with other peers were seen).

Maybe due to that error, Caliper is not able to verify the positive rest of the transactions and mark them invalid, although they are added to the ledger. In the next round of transaction sending, the rate network could add a new block.
- In some cases where transaction failures were 100% peer logs were constantly showing attempts to synchronize ledgers state for multiple peers. For some reason the network could not reach agreement on blocks ordering and remain unsynchronized, therefore the validator could not have added blocks in his private ledger. It might be a bug in fork resolver of Hyperledger Sawtooth v1.0.5 or message exchange protocol.
- Transaction processor responsible for processing transactions might be non-deterministic.
- Block size has a big impact on the performance for Hyperledger Iroha, bigger block size can decrease average latency and increase throughput.
- Analysis of network traffic distributions shows that it is better to equally distribute transactions in the Iroha network.

5. Conclusions

This paper shows the performance evaluation of three different consensus algorithms used in private blockchain networks and implemented in existing solutions. Each chosen consensus algorithm is based on a different principle. Hyperledger Iroha implements the YAC consensus, which is a voting-based Byzantine Fault Tolerant algorithm. Hyperledger Sawtooth implements the PoET algorithm which is proof-based consensus with probabilistic finality. Hyperledger Fabric v1+ abandons the standard state-machine replication protocol and introduces a new execute-order-validate architecture with Kafka Ordering service to improve performance of the platform compared with standard vote-based protocols.

Performance evaluation and scalability assessment were done by varying different sets of parameters such as block size, transaction sending rate, network traffic distribution, and network size. Performance evaluation was done based on average transaction latency, network throughput, and transaction failure rate. Scalability was assessed based on changes in transaction latency and throughput with increasing network size. Test results let to study the impact of a particular parameter on the blockchain network performance and show how they can be adjusted to improve performance. Performance measurements were gathered by Hyperledger Caliper, which is a dedicated benchmark platform for Hyperledger solutions. Hyperledger Caliper does not support the newest Hyperledger platform versions. The main conclusions are as follows: Hyperledger fabric was found to be the most deterministic regarding its performance results. Minimal transaction latency was in all cases between 0.2-0.4 seconds. The average latency of fabric platforms grows linearly with increasing sending rate. Throughput grows to a saturation point after which it is constant. Smaller fabric networks have smaller average latency and higher throughput than bigger networks with the same transaction sending rate. Tests show that adding endorsing peers to the channel decreases throughput and increases the average latency, while adding additional peers without an endorsing role does not have an impact on latency and throughput changes. Network size depends on the number of endorsing peers. Scalability analysis shows that adding new endorsing peers decreases the performance of the blockchain network. Test cases did not show any significant impact of block size on latency or throughput for Hyperledger fabric. The failing transaction might be caused by Caliper timeouts rather than network problems which can not commit the transaction fast

due to high overload. Minimal latency in Hyperledger Iroha according to experiments not depend on network size and transaction sending rate. For all sending rates it was around 4 seconds. Both the minimal and average latency is almost constant and does not depend on the transaction sending rate. Scalability analysis has shown that Hyperledger Iroha network performance worsens for networks which size exceeds thirty nodes. Hyperledger Sawtooth fails in many tests. Possible failures might be the result of: a bug in Caliper SawtoothAdapter used, a bug in PoET CFT implementation of Sawtooth v1.0.5, or bug in transaction processor handling 'Simple' transactions. Due to those failures further investigation must be done. Maximal used CPU power during the test was 15%, all network components were started and managed by a single docker container. It might suggest that the docker could not use all available power to properly manage nodes. Hyperledger Fabric is the most adult platform among all Hyperledger platforms and is used already in many production systems.

References

1. Bamakan, S.M.H., Motavali, A., Bondarti, A.B.: A survey of blockchain consensus algorithms performance evaluation criteria. Expert Systems with Applications 154, 113385 (2020)
2. Caliper, H.: Hyperledger caliper architecture. Electronic Article. url: https://hyperledger. github. io/caliper/docs/2_Architecture. html (visited on 03/10/2019) (2019)
3. Cruz, Z.B., Fernández-Alemán, J.L., Toval, A.: Security in cloud computing: A mapping study. Computer Science and Information Systems 12(1), 161–184 (2015)
4. De Angelis, S., Aniello, L., Lombardi, F., Margheri, A., Sassone, V.: Pbft vs proof-of-authority: applying the cap theorem to permissioned blockchain. In: Italian Conference on Cyber Security. p. 11 pp. (01 2017)
5. Dhillon, V., Metcalf, D., Hooper, M.: The hyperledger project. In: Blockchain enabled applications, pp. 139–149. Springer (2017)
6. Dinh, T.T.A., Wang, J., Chen, G., Liu, R., Ooi, B.C., Tan, K.L.: Blockbench: A framework for analyzing private blockchains. In: Proceedings of the 2017 ACM International Conference on Management of Data. pp. 1085–1100. ACM (2017)
7. Feng, L., Zhang, H., Chen, Y., Lou, L.: Scalable dynamic multi-agent practical byzantine fault-tolerant consensus in permissioned blockchain. Applied Sciences 8(10), 1919 (2018)
8. Gorenflo, C., Lee, S., Golab, L., Keshav, S.: Fastfabric: Scaling hyperledger fabric to 20,000 transactions per second. arXiv preprint arXiv:1901.00910 (2019)
9. Group, H.A.W., et al.: Hyperledger architecture volume 1: Introduction to hyperledger business blockchain design philosophy and consensus (2017)
10. Huang, K., Chen, Y., Jia, H., Lan, J., Yan, X., Wang, Z.: Fast multicast scheme with secure network coding in cloud data centers. Computer Science and Information Systems 13(2), 531–545 (2016)
11. Jovanović, B., Milenković, I., Bogićević-Sretenović, M., Simić, D.: Extending identity management system with multimodal biometric authentication. Computer Science and Information Systems 13(2), 313–334 (2016)
12. Kedziora, M., Kozlowski, P., Szczepanik, M., Jozwiak, P.: Analysis of blockchain selfish mining attacks. In: International Conference on Information Systems Architecture and Technology. pp. 231–240. Springer (2019)
13. Li, W., Sforzin, A., Fedorov, S., Karame, G.O.: Towards scalable and private industrial blockchains. In: Proceedings of the ACM Workshop on Blockchain, Cryptocurrencies and Contracts. pp. 9–14. ACM (2017)

14. Moser, M., Eyal, I., Sirer, E.G.: Bitcoin covenants. In: International Conference on Financial Cryptography and Data Security. pp. 126–141. Springer (2016)
15. Nasir, Q., Qasse, I.A., Abu Talib, M., Nassif, A.B.: Performance analysis of hyperledger fabric platforms. Security and Communication Networks 2018 (2018)
16. Nguyen, G.T., Kim, K.: A survey about consensus algorithms used in blockchain. Journal of Information processing systems Vol. 14, No. 1, pp. 101-128, Jan. 2018 (2018)
17. Performance, H., Group, S.: Hyperledger blockchain performance metrics, https://www.hyperledger.org/HLWhitepaperMetricsPDFV1.01.pdf
18. Rüsch, S.: High-performance consensus mechanisms for blockchains (2018)
19. Sousa, J., Bessani, A., Vukolic, M.: A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform. In: 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN). pp. 51–58 (June 2018)
20. Sukhwani, H.: Performance modeling & analysis of hyperledger fabric (permissioned blockchain network). Duke University (2018)
21. Sukhwani, H., Martínez, J.M., Chang, X., Trivedi, K.S., Rindos, A.: Performance modeling of pbft consensus process for permissioned blockchain network (hyperledger fabric). In: 2017 IEEE 36th Symposium on Reliable Distributed Systems (SRDS). pp. 253–255. IEEE (2017)
22. Sukhwani, H., Wang, N., Trivedi, K.S., Rindos, A.: Performance modeling of hyperledger fabric (permissioned blockchain network). In: 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA). pp. 1–8. IEEE (2018)
23. Thakkar, P., Nathan, S., Viswanathan, B.: Performance benchmarking and optimizing hyperledger fabric blockchain platform. In: 2018 IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS). pp. 264–276. IEEE (2018)
24. Turki, H., Salgado, F., Camacho, J.M.: Honeyledgerbft: Enabling byzantine fault tolerance for the hyperledger platform
25. Vukolić, M.: The quest for scalable blockchain fabric: Proof-of-work vs. bft replication. In: Camenisch, J., Kesdoğan, D. (eds.) Open Problems in Network Security. pp. 112–125. Springer International Publishing, Cham (2016)
26. Wang, R., Ye, K., Meng, T., Xu, C.Z.: Performance evaluation on blockchain systems: A case study on ethereum, fabric, sawtooth and fisco-bcos. In: International Conference on Services Computing. pp. 120–134. Springer (2020)

Arnold Woznica received his M.Sc. degree in Computer Science from University of Science and Technology, Wroclaw, Poland in 2019. His research area of interests encompass blockchain systems and software engineering.

Michal Kedziora received the Ph.D. degree in Computer Science from the Wroclaw University of Science and Technology, Wroclaw, Poland, in August 2014, and the M.S. and Engineer degree in Computer Security from the University of Technology in Wroclaw, Poland, in December 2006. In 2017 he finished postdoc at University of Wollongong, Australia. He was working as a Visiting Researcher at University of Technology Sydney, Australia (2019) and Embry-Riddle Aeronautical University, Daytona Beach, FL, USA (2020).

Received: May 07, 2021; Accepted: January 25, 2022.

Computer Science and Information Systems 19(2):679–707 https://doi.org/10.2298/CSIS210512059P

PE-DCA: Penalty Elimination Based Data Center Allocation Technique Using Guided Local Search for IaaS Cloud

Sasmita Parida[1,2], Bibudhendu Pati[1], Suvendu Chandan Nayak[2], Chhabi Rani Panigrahi[1], and Tien-Hsiung Weng[3]

[1] Department of Computer Science,
Rama Devi Women's University, India
sasmitamohanty5@gmail.com
patibibudhendu@gmail.com
panigrahichhabi@gmail.com
[2] Department of Computer Science and Engineering,
Gandhi Institute for Technological Advancement,
BPUT, Bhubaneswar, India
suvendu2006@gmail.com
[3] Science and Information Engr. (CSIE),
Providence University, Taiwan
thweng@gm.pu.edu

Abstract. In Cloud computing the user requests are passaged to data centers (DCs) to accommodate resources. It is essential to select the suitable DCs as per the user requests so that other requests should not be penalized in terms of time and cost. The searching strategies consider the execution time rather than the related penalties while searching DCs. In this work, we discuss Penalty Elimination-based DC Allocation (PE-DCA) using Guided Local Search (GLS) mechanism to locate suitable DCs with reduced cost, response time, and processing time. The PE-DCA addresses, computes, and eliminates the penalties involved in the cost and time through iterative technique using the defined objective and guide functions. The PE-DCA is implemented using CloudAnalyst with various configurations of user requests and DCs. We examine the PE-DCA and the execution after-effects of various costs and time parameters to eliminate the penalties and observe that the proposed mechanism performs best.

Keywords: Cloud Computing, Data Center, Allocation, Penalty, Meta-heuristic, Guided Local Search.

1. Introduction

Data centers are located through an assortment of topographical regions in cloud computing. As per the user request, the cloud offers types of assistance utilizing the pay-per-use model. Cloud computing offers services through inter-process correspondence between various server farms[1]. For proficient inter-process correspondence, a center level is essential among the clients and cloud providers. The middle person is answerable for simple organization, allotment, and cloud administrations' executives to

the cloud clients. Presently, we essentially center around end clients' vicinity, so service providers guarantee clients' most extreme fulfillment with all availability of assets [2], [3]. Because of the rapid increase in the number of users in the cloud, administrations' requests are rapidly expanded. Accordingly, the cloud upgrades data centers' dependability and accessibility to offer independent services, just as heterogeneous clients [4]. The best DC selection is to oblige solicitations of a specific client. Furthermore, adjusting the heaps among the DCs ignoring the expense is also a monotonous issue [5].

The allocation of on-demand resource allocation initiates a few pre-requisite steps. It must be followed for better resource allocation, such as request processing, searching for data centers, allocating on-demand resources, computing, monitoring, and releasing resources [6]. The request processing phase analyzes the on-demand parameters such as the number of users, number of cluster nodes, request size, the demand of storage, CPU, memory, number of processors, type of operating systems, and several others on the cloud service provider [7], [8] then locating suitable data centers to fulfill the user requirements with minimum resources where the requests can be executed [9]. The network latency has much more impact in allocation which can be reduced if multiple instances will be deployed near by the users [10]. However, the searching mechanisms consider the basic parameters related to cost, time, Service Level Agreements (SLAs), and Quality of Service (QoS) [11], [12]. The resource allocation process occurs, keeping task scheduling and load balancing and initiating the computing process. The monitoring process keeps track of the computing process and resources; whenever the computing process is over, the allocated resources get released and ready to further allocation [13]. Among all these processes, the searching of data center processes favourably impact allocation. Eventually, it is a tedious task during peak hours to refer to the suitable DCs. Thus, the selection of a suitable DC for the on-demand request is a challenging aspect. Service broker policy routes the user requests after finding suitable DCs for resource allocation and keeps load balancing [14], [15].

In the last decades, various meta-heuristic and optimization-based DC allocation mechanisms have been proposed. These mechanisms target in optimizing cost and time, managing SLA and QoS by computing optimal DC lists. While computing the optimal DC list, researchers have focused less on the importance of searching techniques. Assume is the set of solution (DC list) computed through some techniques for a set of variables (user request) . Let f(x) computes the optimal solution si for user request RJ. While considering multiple parameters may not be an optimal solution for RJ. The optimal solution for RJ could be possible if we compute the penalty associated with the parameter set P. In cloud computing, if the data center DCi is selected for Rj, then the penalty for users' set for the parameters needs to be computed.

Motivation and Contribution. The DC allocation techniques need the discussion of search techniques. Local Search (LS) can be considered as the right solution with less time. However, out of all neighbors present in the search space, LS can be trapped as local optima- position. Over the year, different approaches are suggested for the improvement of the LS effectiveness. Simulated Annealing (SA), Tabu Search (TS), and Guided Local Search (GLS) are providing the supports to improve LS rather than the local optimum. The GLS technique is a meta-heuristic and a global optimization algorithm that utilizes an embedded LS algorithm. It is an expansion to the LS

algorithm, and for example, Hill Climbing is comparable in the system to the Tabu Search calculation and the Iterated Local Search calculation.

GLS is a penalty based meta-heuristic searching mechanism to solve issues related to local minima due to augmented objective function. GLS is used to deal with combinatorial optimization problems by improving efficacy of local search process. It calculates utility for each penalized feature. The basic aim is to assign penalties to all those features in the search space having high cost function values with maximum utility. While searching the suitable DC for allocation, the local search does not consider the penalty and utility. The local search techniques compute the searching with the constraints. The DC selection requires multiple parameters to select the suitable one. In case of random DC allocation, the penalty may be more with less utility. So the work suggests implementing GLS technique in the DC allocation mechanism.

In this work, we propose a new DC allocation technique named as PE-DCA, using the meta-heuristic search technique GLS. The PE-DCA mechanism allocates suitable DCs for the on-demand user requests by evaluating the penalty associated with time and cost metrics. The total cost of the VM is minimized, and the response time and processing time are reduced. The contributions of this work are mentioned as follows.

- Propose a novel meta-heuristic DC allocation mechanism for on-demand resources.
- Address the penalty associated with searching for a suitable DC.
- Formulate and discuss the proposed GLS based DC allocation technique (PE-DCA)
- Study the performance of the proposed mechanism through the java-based simulation tool called CloudAnalyst.
- Compare the performance parameters such as total cost, response, and processing time with the existing techniques.

The remainder of this work is organized as follows. Section 2 discusses the formulation of the problem statement and discusses the importance of the searching technique in allocation, while the background of the GLS technique is highlighted in Section 3. The detailed presentation of the proposed PE-DCA mechanism model and algorithm is discussed in Section 4. The simulation and the performance results are discussed in Section 5 to present the importance of PE-DCA in allocation. Section 6 presents the performance comparison results with the existing mechanisms for allocation in clouds. The related work for resource allocations are presented in Section 7, and finally Section 8 summarizes the contribution of the proposed work and the future directions.

2. Problem Statement Formulation

This section asserts DC allocation for on-demand resources for which the work is proposed. The importance of searching techniques and their improvement is addressed to select suitable DC. It also presents how the searching approaches can be improved to find globally optimal solutions from the local optimal solutions. The Table.1 presents the symbols and notations with descriptions used in this proposed work.

Table 1. Symbols and Notations

Symbols	Explanation
DC	Data center
PU	Physical machine Unit
VM	Virtual machine
R	Region
Ti	Time
C	Network constant
DTcost	Data transfer cost
VM_{total}	Total VM cost
VM_{cost}	VM cost
ST_{cost}	Storage cost
M_{cost}	Memory cost
α	Regularization parameter
ND_{Time}	Network delay
Res_{Time}	Response time
Req_{in_time}	Request initiated time
D_R	Data size per request in bytes
$BW_{avaliable}$	Available bandwidth
N_{dl}	Network latency due to delay
η	Data center request size
d	Delay parameter
Pro_{time}	Processing time
$RVM_{capacity}$	Request VM capacity
DC_{load}	Load of DC
$DC_{capacity}$	Capacity of DC

Cloud infrastructure is made up of various DCs, and all DCs are integrated with varying configurations and geographical locations are shown in Figure 1. These configuration parameters need to be considered in DC allocation. Let $R = \{R_1, R_2, \ldots R_n\}$ be the set of regions, $DC = \{DC_1, DC_2, \ldots DC_m\}$ be the set of DCs and $\forall DC_i \in R_j$, where $i, j \in n$. Each DC_i consists of a set of physical machine units (PU): $DC_i = \{PU_1, PU_2, \ldots, PU_n\}$ and PU_i represents with a set of VMs: $PU_i = \{VM_1, VM_2, \ldots, VM_n\}$. A set of objective function $F = \{F_1, F_2, \ldots, F_z\}$ can be enhanced. In general, time and cost parameters have an essential role in the resource allocation mechanism. A data center matrix DC_i can be computed with time and cost given in Eqn. (1).

$$DC_i(T_i, C_i) = \begin{bmatrix} T_1C_1 & \cdots & T_1C_n \\ \vdots & \ddots & \vdots \\ T_nC_1 & \cdots & T_nC_n \end{bmatrix} \quad (1)$$

The regions are separated geographically, so the $DC_i(T_i, C_i)$ can vary due to the parameter α, where α stands for data transfer cost and network constant C and is given as in Eqn. (2).

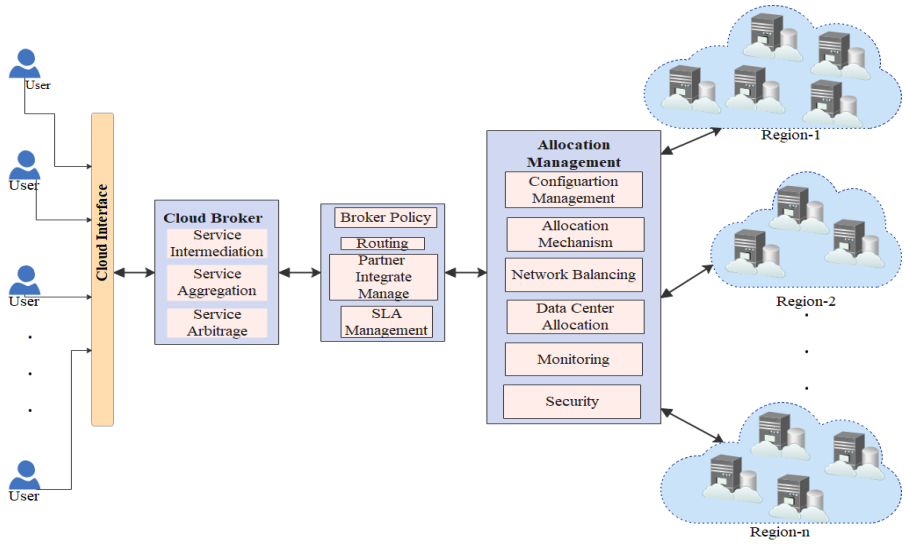


Fig. 1. Cross-cutting of Data Center Allocation

$$R_i(DC_i) = \alpha * DC_i(T_i, C_i) + C \qquad (2)$$

Consider the set of user requests $U = \{U_1, U_2, \ldots, U_m\}$. The on-demand resource of user Ui can be available in multiple data centers in various locations. The challenge is to select the best data center for Ui. All the optimal data centers are not the global optimal data center for Ui due to penalty and utility. The penalty and utility are reciprocal to each other and are influenced by α and C for a data center. It is essential to search the global optimal data center from the local optimal data center list and is shown as in Figure 2.

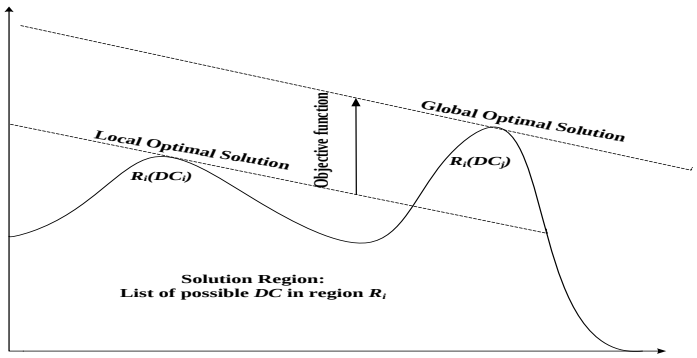


Fig. 2. Optimal Solution Approach

At this point, we have to follow combinatorial optimization as defined by a pair (D,c), where

D = the set of feasible solutions.

c = objective function, which maps each member with D.

The objective is to find a solution d in D that minimizes the objective function c and is formulated as:

$$Min(d)\ \forall\ d \in D \qquad (3)$$

Now, we define a neighborhood N for the pair (D,c) that is covers D and is given by

$$N \mid D \rightarrow 2^d$$

Consider *N(d)* as the neighborhood of *d*, and it has all possible solutions which are reachable from *d* with a single move. Here, the movie is an operator that transforms one form of solution to another with some modification.

Let *m* be the solution called as a local minimum with respect to *N(d)* if and only if

$$c(m) \leq c(d) \mid \forall d \in N(d) \qquad (4)$$

In the process of local searching primarily, we are minimizing the cost function *f* iteratively in various steps, and the current solution *m* being replaced by a solution *y* such that

$$y=c(d) < c(m) \mid d \in N(d) \qquad (5)$$

3. Guided Local Search (GLS)

This section presents the meta-heuristic-based GLS technique. It discusses the derived cost function for GLS in DC allocation, and the penalty function is formulated to find the different penalties associated with various allocation parameters.

To apply GLS, solution features must be characterized for the given problem. The solution features are characterized to recognize arrangements with various qualities, so areas of comparability around local optima can be perceived. The arrangement highlights' decision relies upon objective function and a limited degree on the local search algorithm[16]. The main idea to design the GLS algorithm system is to utilize punishments to empower a local search method to get away from local optima and find the global optima [17]. A local search algorithm is executed until it stalls out in local optima. The local optima highlights are assessed and punished, the consequences of which are utilized in an enlarged cost function utilized by the local search methodology. The local search is rehashed on various occasions utilizing the last local optima found and the enlarged cost function that guides investigation away from arrangements with highlights present in found local optima [18].

GLS is a meta-heuristic search technique[16] that sits on top of the local search technique to change its computation. It is an intelligent searching technique for a combinatorial optimization problem. The main idea states that the technique does iterative use of local search, gathers information from various sources and guides the local search towards a promising and suitable search space. It provides reasonable approximated solutions from local solution space due to its iterative searching mechanism. GLS follows a penalty-based mechanism that leads the local search technique through the guide function, which considers a feature where the cost and penalty are associated. The objective function *f(.)* defines the cost over the feature *i*. GLS develops punishments during a search. It utilizes penalties to assist the local search algorithm by looking through local minimal and plateaus. When the given local search

algorithm settles in a local optimum, GLS changes the objective function by utilizing a particular plan [19].

At that point, the local search utilizes an expanded objective function intended to bring the search out of the local optimum. The key idea is to change the objective function. GLS has been applied to a non-insignificant number of issues and found to be productive and viable. It is generally easy to execute and apply, with scarcely any parameters to tune. Assume the objective function *f(.)* and the guide function *g(.)* is treated as augmented objective function with the feature *i* and the corresponding penalty p_i in the solution space *s*. In each iteration *g(.)* contributes as a guide function and adjusts the increasing the penalties for the feature *i* from *s* and its utility can be evaluated. The feature *i* with the highest utility value is computed from *s*.

The greedy approach is followed in local searching techniques where we start with random solution space and stop with local minima. Here, instead of searching the whole solution space, we consider approximated solutions space using the iterative approach to compute the optimal solution.

In GLS, the general local search is given by the form as in Eqn. (6).

$$d_2 \leftarrow DCLocalSearch\ (d_1, c) \qquad (6)$$

where, d_2 is the local minimum and d_1 is the initial solution, and *c* is the objective cost function.

GLS defines solution feature as an augmented function, here we consider cost function as it and put the non-trivial solution element in the solution feature. Due to this property, each feature solution depends on the problem and interfaces within a particular application. The cost may affect directly or indirectly the solution feature.

A feature f_i *is* defined in Eqn. (7) as

$$I_i(d) = \begin{cases} 1, & if\ the\ solution\ has\ property\ i \\ 0, & Otherwise \end{cases} \quad and \quad d \in D \qquad (7)$$

The constraints on features are given by augmenting the cost function *c* to set a penalty.

Now, we have a new cost function called augmented cost function *g(d)* and is given in Eqn. (8).

$$g(d) = c(d) + \delta \sum_{i=1}^{n} P_i \,.\, I_i(d) \qquad (8)$$

where *n* = Number of features defined over solutions.

P_i= Penalty parameter for f_i

δ = Regularization parameter

Here, δ relates to the solution cost and has an impact on the search process and defined as in Eqn. (9).

$$\delta = {DC_{load}}/{DC_{capacity}} * VM_{load_i} \qquad (9)$$

A penalty vector is given over the defined solutions throughout the search process. During each iteration, the local search finds a local minimum over the possible set of solutions and let the penalty vector is given by $P = (P_1, P_2 \ldots P_n)$.

4. Proposed DC Allocation Mechanism (PE-DCA)

The detailed design and working mechanism of the proposed PE-DCA are described in this section. Various formulated functions followed by the diagrammatical representation of the proposed PE-DCA is also presented.

4.1. System Model

In this work, we propose a data center allocation mechanism named as PE-DCA using a meta-heuristic GLS technique. The local search does not consider the penalty for on-demand resources and multiple feasible data centers are also possible in different regions. The PE-DCA considers the penalty associated with feasible data centers. We derive the cost function *C( )* and time function *T( )* to consider different associated parameters. The total VM cost (VM_{total}) can be computed by VM cost (VM_{cost}) and data transfer cost (DT_{cost}) and is given as in Eqn. (10).

$$C(VM_{total}) = C(VM_{cost}) + C(DT_{cost}) \quad (10)$$

where, VM_{cost} is calculated as the sum of the cost per VM (VMP_{cost}), memory cost (M_{cost}), and the storage cost (ST_{cost}) and is given as in Eqn. (11).

$$C(VM_{cost}) = C(VMP_{cost}) + \sum_{i=1}^{x} C(M_{cost}) + \sum_{i=1}^{y} C(ST_{cost}) \quad (11)$$

where *x* defines the number of required memory units for MB main memory and *y* signifies the required storage units to X-MB [20][21]. These values are defined by the service provider during DC configuration using different pricing models [22]. DT_{cost} is derived by utilizing Eqn. (12) from available bandwidth ($BW_{avaliable}$), Data Size Request in bytes (D_R), and the number of PU as:

$$C(DT_{cost}) = \sum_{i=1}^{PU} \left(\beta * \frac{D_R}{BW_{avaliable}}\right) \quad (12)$$

The VM_{cost} is fixed in the feasible solution space, though the DT_{cost} is varied from the data center to the data center. So, VM_{total} also has impact in searching for the optimal global solution. We also compute the response time utilizing the Eqn. (13) and consider its importance in searching for the optimal solution.

$$T(Res_{Time}) = \left(c_1 + T\left(t_i - Req_{in\_time}\right)\right) + \left(\frac{D_R}{BW_{avalaible}} + ND_{Time}\right) + c_2 \quad (13)$$

where Res_{Time}= Response time, c_1= Time required to take decision for allocation, t_i=tentative start time of allocation, Req_{in_time}=Request initiated time, ND_{Time} = network delay and c_2 = Time required for configuration checking. The proposed architecture is shown as in Figure 3.

The PE-DCA finds the optimal data center for the on-demand request. If the searched data center is overloaded, then PE-DCA refers to the next optimum data center within the same region. In a region when all the data centers are allocated, then PE-DCA searches the next closest region.

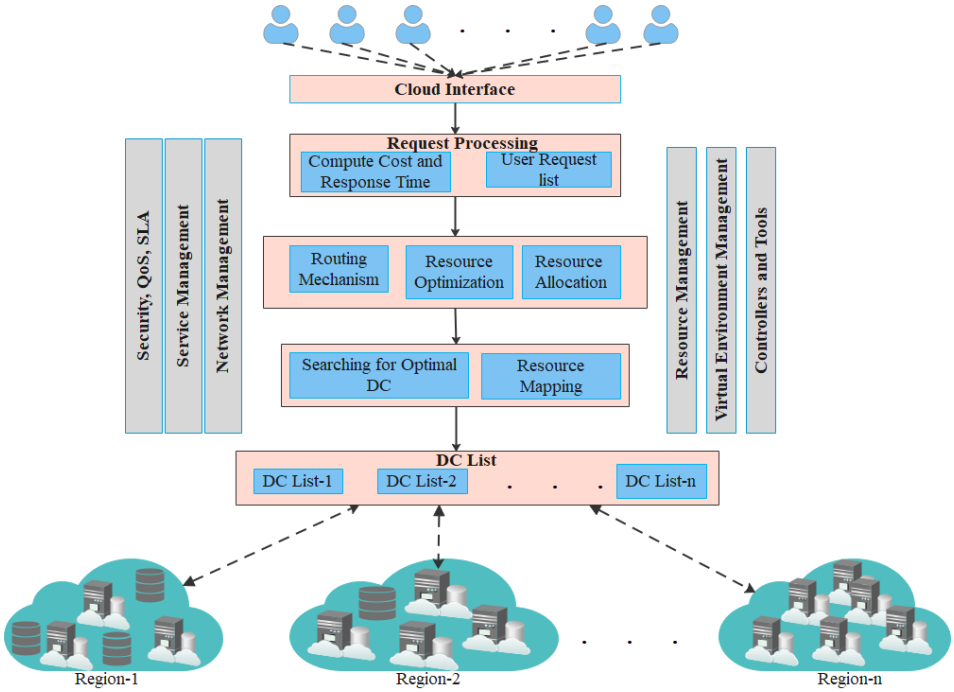


Fig. 3. Proposed PE-DCA Architecture

The proposed mechanism considers the network latency due to delay (N_{dl}) along with cost and time to search a suitable data center. We can compute N_{dl} as the time delay for a request from its initiation to retrieve a processing response. The latency directly depends on the $BW_{avaliable}$ and the delay parameter (d) and cooperatively impacts the response time, though discursively increase the overall cost for the VM.

$$N_{dl} = d + \eta\,[request/job]/BW_{avaliable} \quad (14)$$

$$\min Res_{Time} \propto \min N_{dl} \geq \min VM_{total}$$

$$\min Res_{Time} \propto \max BW_{avaliable}$$

$$\min VM_{total} \propto \min Pro_{Time}$$

where η = Data center request size and Pro_{Time} = Processing time.

The bandwidth alludes to the amount of data that can be conveyed inside the network from users to various DCs farms at a time instance. Likewise, it fundamentally influences the required response time of the request. Besides, the response time and all-out expense of the cloud environment proportionately influence the response time. Appropriately, the data centers farm can deal with more demands within a time unit when the processing time reduces, *i.e.*, roughly with an improved response time. Thus, the cloud framework's final cost decreases with processing time, resulting in refined overall performance.

4.2. Penalty Elimination and Allocation Technique

The proposed PE-DCA allocates the on-demand resources on the computed values of cost and time and is defined in Eqn. (10) and (13). The network latency as defined in Eqn. (14) is countered with cost and time in the suitable data center's searching mechanism. The various computational steps of PE-DCA are highlighted in Figure 4. The basic principle of GLS is analyzing constraints on solution features. For each iteration, the local search tends to a local minimum, and the penalty parameter defining one or more features over different solutions can be incremented by GLS. As the penalty parameter gradually increases for one or more features over a possible solution set the increment of penalty signifies that the penalized feature must be avoided by local search. Thus, high-cost features have more penalties in comparison to low-cost features.

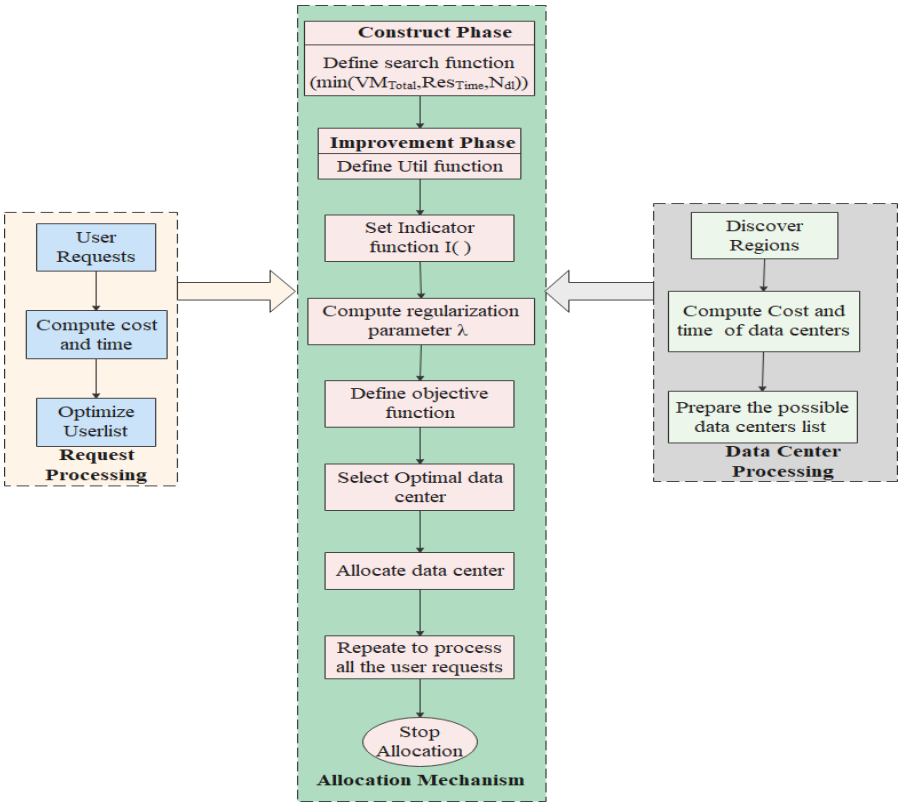


Fig. 4. Proposed PE-DCA Model and Design Steps

We assume that all the penalty values for all feature is set to 0 initially. Let each feature f_i is assigned to a total cost C_i and is represented by a vector as given in Eqn. (15).

$$C(d) = (C_1, C_2 \ldots C_n) \mid \forall C_i \geq 0 \qquad (15)$$

If the feature f_i is found as a local minimum solution D_* then

$I_i(D_*) = 1$ as per Eqn. (7).

Let a vector *L* considered as indicator function values which keep local minimum D_* and is represented as in Eqn. (16).

$$L(D_*) = \left(I_1(D_*), I_2(D_*) \dots I_n(D_*)\right) \quad (16)$$

There is a utility function related to each data center's possible solutions. In local minimum (D_*), the penalty values are increased by one if the utility function is maximized for each feature f_i.

$$DC_{Util}(D_*, f_i) = I_i(D_*) \cdot \frac{C_i}{1 + P_i} \quad (17)$$

The DC with minimum utility is selected before being assigned in a local minimum. Thus the utility function uses Vector *L(D∗)* and cost vector *C*. We introduced DC utility associated with each instance specified in Eqn.(17) to avoid the penalty incorporated to be of high cost. The PE-DCA evaluates the during allocation. It is calculated using Eqn. (18).

Request VM capacity ($RVM_{capacity}$): $RVM_{capacity} = P_s * N \quad (18)$

where P_s= processing speed of the processor in MIPS and *N*=Number of processors.

VM Taskload (VM_{load}): The VM_{load} states the workload of each VM with service rate at a time *t*. So the VM_{load} for i^{th} VM at time *t* is calculated by using Eqn. (19).

$$VM_{load_{i,t}} = k * {TL_i}/{X\left(VM_{i,t}\right)} \quad (19)$$

where k=1,2, …, n, TL_i = Task Length in Million Instructions (MI) and $X(VM_{i,t})$ = service rate of VM_i at time *t*.

Datacenter load (DC_{load}): It represents the number of the task of i^{th} VM in a request for a data center and is computed by using Eqn. (20).

$$DC_{load} = \sum_{j=1}^{m} pVM_{i,t} \quad (20)$$

where *p* = number of tasks associated with the request.

Datacenter capacity ($DC_{capacity}$): The DC capacity represents the sum of VM capacity and is calculated by using Eqn. (21).

$$DC_{capacity} = \sum_{j=1}^{m} RVM_{capacity} \quad (21)$$

Expected processing time (EPT): It is the time counted before allocation to complete the task with the on-demand request in the data center. EPT is defined as a total load of a data center and the total capacity and is given as in Eqn. (22).

$$EPT = {DC_{load}}/{DC_{capacity}} \quad (22)$$

Algorithm1:DC_Guided_Local_Search(P,g(d),α,[$I_1(D_*), I_2(D_*) \dots I_n(D_*)$],

[$C_1, C_2 \ldots C_n$],n)
Inputs: dcList=[*1,2,…,n*]
Output: **D**$_*$

1 Begin
2 k←0
3 D_0←InitialAllocation(P)
4 Set all penalties cost to 0
5 **for** *i*←*1* to *n*
6 P_i←0
7 $$g(d) = c(d) + \alpha \sum_{i=1}^{n} P_i \cdot I_i(d)$$
8 **While** stoppingCriteria==false
9 do
10 D_{k+1}←DClocalSearch(D_k,g)
11 **for**←*1* to *n* do
12 $$DC_{Util}(D_*, f_i) = I_i(D_*) \cdot \frac{C_i}{1 + P_i}$$
13 **for** each *i* such that DC_{Util_i} is maximum **do**
14 P_i←P_i+1
15 End for
16 k← k+1
17 End for
18 End while
19 *dcName*=D_*← best solution with the minimum of DC_{total} and Res_{Time}
20 Return D_*
21 End for

The proposed mechanism PE-DCA implements two procedures *DC_Guided_Local_Search() and Select_DC()*. The *DC_Guided_Local_Search()* empowered with the meta-heuristic search mechanism GLS. The design approach follows the *InitialAllocation()* to construct the initial solution for *D* over problem *P*. The *DClocalSearch(Dk,g)* searches local minimum and improves the solution with D_k and compute the utility till the *stoppingCriteria* fails. By eliminating the penalty, the best solution is selected based on the objective function for DC_{total} and Res_{Time} as defined in Eqn. (10) and (13). The procedure *Select_DC()* emphasizes in selecting the data center with minimum penalty. It computes the cost and time using the defined objective functions for each user request to penalty while allocating the data center. The minimum penalty of cost and time defined in step 7 is the selected data center's parameters. For allocating the computing data center, we compute DC_{load} and $DC_{capacity}$ as defined in Eqn. (20) and (21) to know whether the computed data center is a suitable one or not; if suitable, the *dcName* is tracked with the index for allocating the user's resources.

Algorithm 2:**Select_DC( )**
Inputs: dcList=[*1,2,…,n*], regionList=[*1,2,…,m*], where *m*<*n*
Output: **dcName[]**

Begin
For the selected region get data center index *regionList*
Get regionalDatacenterIndex.get(region)
Keep region list for selected data center
if *regionList* is not *NULL* then

listSize_size(regionList)

if *listSize_size* is 1 then
dcName_regionList.get(0)
else

for i=1 to n

Compute $C(VM_{total})$ and $T(Res_{Time})$for each DC

Create the list p for DC with $min(VM_{total}, Res_{Time})$

End for

for all *p*

Compute $DC_{capacity}$and DC_{load}

if (DC_{load}< $DC_{capacity}$)

get DC index and prepare *dcName[ ]*

dcName[index]=dcName_regionList.get(p)

End if

End for
End if else
End if
Return dcName[ ]
End

5. Simulation and Results

This section presents the proposed PE-DCA outcomes, briefing the required simulation tool, setup, and configuration, and evaluating different parameters such as cost, response time, and processing time obtained through simulation and analyzes the performance. CloudAnalyst [20] is an open-source Java-based cloud tool issued to study the proposed PE-DCA behavior to support a built-in environment.

5.1. Simulation Environment

The standard parameters of CloudAnalyst were customized to examine the results of PE-DCA. CloudAnalyst defines six geographical regions indexed as ($R_0, R_1, \ldots, R_5$) to locate DCs in its simulation environment. We consider various network delays in milliseconds among the simulation regions and is given as in Table.2. We assume a minimum network delay of 25 milliseconds in a similar region and vary to a maximum of 500 milliseconds for other regions during our simulation. We consider various bandwidths ranging from 800 to 2500 Mbps for data transmission among the regions and are given in Table.3 in our simulation study. PE-DCA is examined for a different set of user requests and DCs and is presented in Table.4.

Table 2. Network delay in regions (ms)

Region	R_0	R_1	R_2	R_3	R_4	R_5
R_0	25	100	150	250	250	100
R_1	100	25	250	500	350	200
R_2	150	250	25	150	150	200
R_3	250	500	150	25	500	500
R_4	250	350	150	500	25	500
R_5	100	200	200	500	500	25

Table 3. Bandwidth in regions (Mbps)

Region	R_0	R_1	R_2	R_3	R_4	R_5
R_0	2000	1000	1000	1000	1000	1000
R_1	1000	800	1000	1000	1000	1000
R_2	1000	1000	2500	1000	1000	1000
R_3	1000	1000	1000	1500	1000	1000
R_4	1000	1000	1000	1000	500	1000
R_5	1000	1000	1000	1000	1000	2000

Table 4. Experimental User and Data Center Set

User Set	No of Users	No of DCs
1	100	28
2	150	43
3	200	55
4	250	68
5	300	80

The network bandwidth and transmission delay for the considered experiments and scenarios were kept constant during the simulation. The proposed simulation model is shown in Figure 5 and the user requests are generated using the user base interface of CloudAnalyst. Similarly, the data center configuration interface of CloudAnalyst is used to configure the data centers. Here, we created two new classes under the package of *cloudsim.ext* for implementing PE-DCA, and those classes along with the package are imported to the package *cloudsim.ext.gui*. Finally, all the classes are executed from the caller's primary method *GuiMain.java* class. The Java-based application is developed to implement the computed functions and is shown in Figure 5.

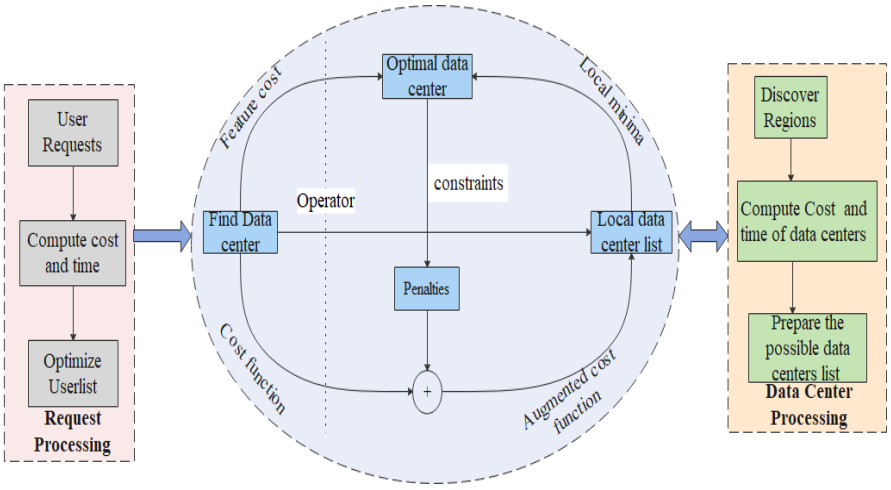


Fig 5. Implementation Model of PE-DCA for Simulation

Here, we consider cost and time to prepare the user list and derive total VM cost in Eqn. 10 and response time in Eqn. 13. The user list is computed with minimum total VM cost and response time. Similarly, we prepare the DC list considering the compute cost and response time. As we earlier discussed GLS work on the top of the local search technique in incremental way. We implement the cost and the response time function as the augmented function for incremental growth of the searching technique. The java-based function is developed which consider the 1st user request from the user list and estimate the cost to the corresponding DC list. This process goes incrementally and computes the penalty and utility corresponding to the DCs. The similar process is repeated for the response time augmented function. The application keeps all the records of the penalties and utilities of each user request with respect to the DCs. Here, we consider DC_{load} and $DC_{capacity}$ as the searching constraints. If DC_{load} is less than $DC_{capacity}$ then the proposed system selects the minimum penalty and returns the DC index for allocation.

6. Performance Evaluation

The study of PE-DCA is performed by considering diverse userbase (user request) and data center configuration specifications. The performance was obtained for a variable number of user requests, with a maximum of up to 300 in different scenarios. The number of user requests per hour varies from 60-90, the request size is set to 1-500 KB, and the average off-peak user is 50 and 100. The data center is configured as x86 architecture, Linux OS, Xen VMM with variable physical H/W units (1-4) with four numbers of processors, each having 10,000 MIPS. We consider different standard costs for the data center: the cost per VM per Hr as 0.1$, memory cost as 0.05$, and data transfer cost per GB as 0.1$. The performance is studied by considering a number of data centers vary between 28-80, which is less than 30% of the number of user requests in different scenarios.

The performance of PE-DCA is also studied under different scenarios to determine suitable data centers for a user request. We implement the derived function (Eqn.(10)) for VM_{total} and Res_{ume} for each user request (Eqn. (13)) in the set. The DC_{util} function as in Eqn.(17) is computed to examine the utilization for each data center before allocation. Implementing DC_{util} is to maintain load balancing among the DCs and is used as the *stoppingCriteria* in PE-DCA. We compute the response time penalty in millisecond and cost penalty in dollar ($). However, from the obtained results it was found that the penalty in allocation increases the response time, and the penalty in cost improves the total VM cost. So, as the penalty is reduced for time and cost, the response time and total VM cost are reduced, and better data center allocation is achieved.

PE-DCA evaluates the deviation in the cost for each user request for the DCs. On the other hand, we had also evaluated the response time deviation for each user request and the selected DCs. The penalties may vary for each set of the user request and the DC. So, we evaluated the penalty of cost and response time in average and is noted in Table.5 for users set. Figure 6 depicts the response time penalty associated with a different set of user requests. It signifies that the deviations in cost and time for each user request need to be eliminated. Note that the increase in deviation may increase the total cost and response time. The minimum and maximum response time penalty of 0.02 ms and 0.8 ms were observed during the simulation. It was noticed that the average penalties varies between 0.6025 ms to 1.0575 ms during the experimentation. The cost ($) penalty associated to user requests in different user sets is shown in Figure 7, where the minimum and maximum cost penalties are 0.02$ and 0.28$ and the simulation studies consider the average cost penalties within the range 0.1231$ to 0.1394$ which are responsible for increasing the total VM cost and need to be eliminated.

Table 5. The Cost and Response Time Penalty

User Set	Average *P(c)* in $	Average *P(t)* in ms
1	0.1231	0.6025
2	0.1338	0.8583
3	0.1352	0.8709
4	0.1371	0.9732
5	0.1394	1.0575

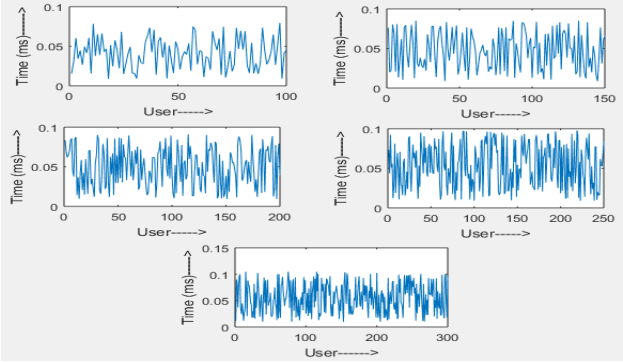


Fig. 6. Computed Response Time Penalty (ms) to User Request in Different User Set

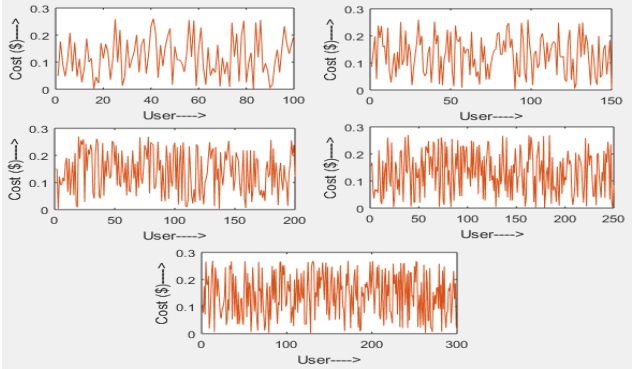


Fig. 7. Cost Penalty ($) associated to User Request in Different User Set

7. Results and Discussion

This section presents simulation results along with the analysis of results. The simulation results of the total cost, response time, and data center processing are discussed for various users. The advantages of the PE-DCA for data center allocation are also highlighted.

7.1. Overall Response Time

The response time computed during the simulation addresses the time required to allocate the data center. It can vary from data center to data center for various bandwidth and network delays. The computation of response time was computed with regard to bandwidth and network delays are given in Table.2 and 3. The response time penalties

occur due to improper allocation, which may be considered as the deviation during allocation. We derived the function for response time (Eqn.(13)) and computed it by considering the penalty associated with each user request. The suitable data center was selected by evaluating the δ defined in Eqn.(9) which examines whether the data center can accommodate the request or not. To evaluate δ, we implemented Eqn. (19), (20), and (21) and computed the minimum, maximum, and average response time for all the user sets are noted in Table.6. It was observed from obtained simulation results that the proposed PE-DCA mechanism requires a minimum of 46.0875677158 ms and a maximum of 173.990628155 ms to allocate the data center. The overall response time for various user request set is monitored and shown in Figure 8.

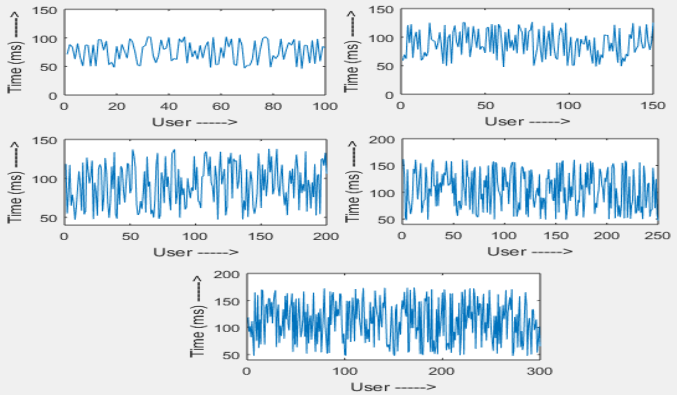


Fig. 8. Overall Response Time (x-axis: Number of User Request, Y-axis: Response Time (ms))

Table 6. Overall Response Time for Various User Sets

User Set	Over All Response Time		
	Minimum	Maximum	Average
1	46.09	102.25	76.28
2	47.74	126.41	88.85
3	47.08	138.05	92.09
4	47.32	161.83	104.67
5	47.34	173.99	111.52

7.2. Overall Total Cost

The CloudAnalyst simulator supports for evaluating different costs such as VM cost, data transfer cost, and total cost. The total cost is the sum of VM cost and data transfer cost. We implemented Eqn. (11) and (12) to evaluate the total VM cost. It can be varied from request to request due to the Request VM capacity defined in Eqn. (18). We computed each request VM capacity and the corresponding cost. We computed the penalty cost associated to user request is shown in Figure 7. In PE-DCA, we observed that the penalty cost was gradually decreasing to a minimum level and was not changed

further as shown in Figure 9. We considered the minimum penalty cost for computing the overall cost metric: VM cost (VM_{cost}), data transfer cost (DT_{cost}), and total cost (VM_{total}) for all user sets and were recorded in Table.7.

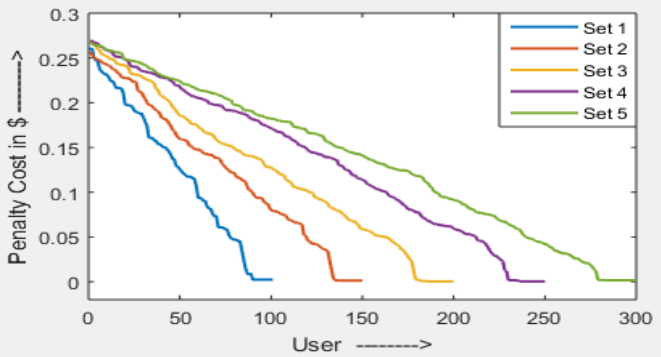


Fig. 9. Penalty Cost Reaching to Minimum Level

Table 7. Overall Cost Metric in $

User Set	VM_{cost}	DT_{cost}	VM_{total}
1	71.73	53.57	125.30
2	84.26	71.48	155.74
3	114.44	83.67	198.11
4	141.76	94.67	236.43
5	176.78	108.54	285.32

7.3. Overall Processing Time

During simulation, we computed the overall processing time for all the user sets. The processing time varies for different types of users due to the on-demand of resources. For each user request, we computed the EPT as defined in Eqn. (22) and considered min(EPT) in the implementation to evaluate the overall processing time. As the proposed work was simulated using the CloudAnalyst tool, we used the in-built mechanism and examined the overall processing time for our configured data centers and user requests rather than defining any procedure. The overall processing time for various user sets is shown in Figure 10, and noted the minimum and maximum time as 3.98 ms and 1371.28 ms in our simulation. The minimum, maximum, and average computational overall processing time was recorded as given in Table.8.

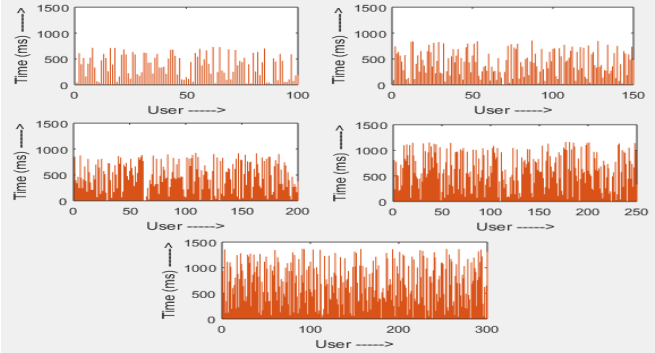


Fig. 10. Overall Processing Time (X-axis: Number of user requests, Y-axis: Processing time (ms))

Table 8. Overall Processing Time (ms)

User Set	Minimum	Maximum	Average
1	19.98	730.47	371.5
2	3.98	855.17	427.99
3	5.05	926.09	475.38
4	4.48	1162.04	607.31
5	6.03	1371.28	692.79

8. Comparison

In this section, we represent the performance comparison of the proposed PE-DCA with different prevailing techniques. We compare the in-built broker policies of CloudAnalyst first, followed by comparing various techniques as proposed researchers for data center allocation.

8.1. PE-DCA vs Benchmark Broker Policy

We compare the performance parameters such as total VM cost, overall response time, and overall processing time for various user sets with the benchmark broker policies of CloudAnalyst. The *Closest Data Center* [20] broker policy focuses on routing user requests based upon the nearest data center. It does not consider other computing parameters for allocation. It allocates the data center by considering the distance among the data center. CloudAnalyst defines *Optimize Response Time* [20] broker policy to optimize the response time of the user request. This policy allocates data centers based upon minimizing the response time of the request. The comparison of average total cost in $ is noted in Table.9.

Table 9. Comparison of Average Total Cost in $

User Set	Closest DC Policy	Optimize Response Time	PE-DCA
1	1.3593	1.2984	1.2530
2	1.1320	1.1471	1.0382
3	1.0332	1.0957	0.9905
4	1.0629	1.1051	0.9457
5	1.0811	1.0444	0.9510

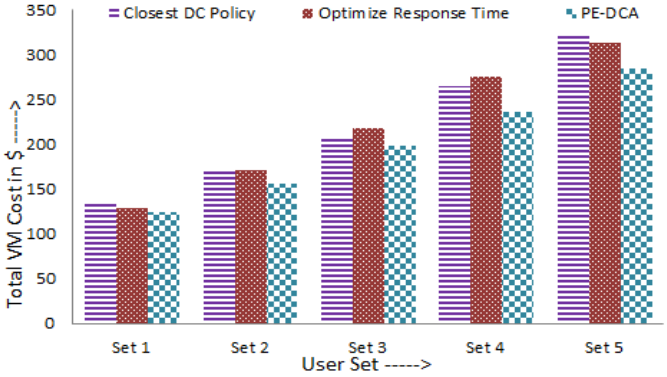


Fig. 11. Comparison of Total VM Cost for User Sets

Fig. 12. Comparison of Average Overall Response Time for User Sets

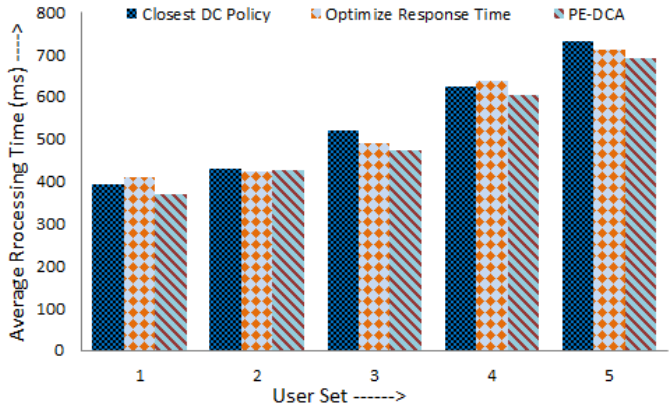


Fig. 13. Comparison of Average Overall Processing Time for User Sets

8.2. PE-DCA vs Existing Approaches

Many researchers have recently used optimization techniques for resource management, mainly the Particle Swarm Optimization (PSO) impacted more. We compare our proposed work with the existing PSO-based mechanisms for cost and response time optimization in resource management for cloud computing. To establish a comparison with existing mechanisms, we develop new java classes under the package of *cloudsim.ext* and import them to the package *cloudsim.ext.gui*. To compare costs, cost-aware PSO (CA-PSO) [23], novel PSO (NPSO) [24], and Modified PSO (MPSO)[25] are considered, where such techniques are executed, and the number of iterations is similar to the number of user requests in the set. The comparison of total cost between the above approaches and proposed PE-DCA is shown in Figure 14, and the average overall response time is depicted in Figure 15. Note that the total VM cost of PE-DCA and existing NPSO and MPSO are found to be approximately equal as shown in Figure 14, but the average overall response time is found to be more than PE-DCA as noticed in Figure 15.

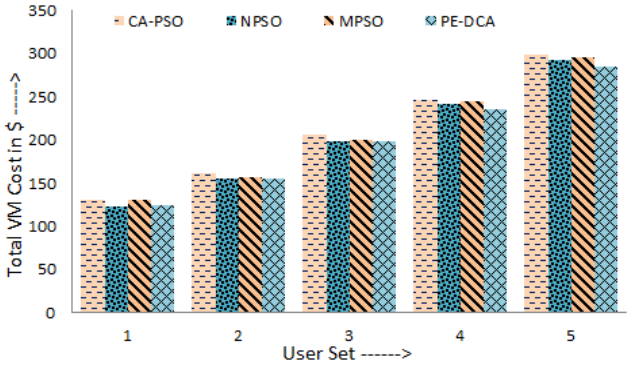


Fig. 14. Comparison of Total VM Cost with PSO Based Techniques

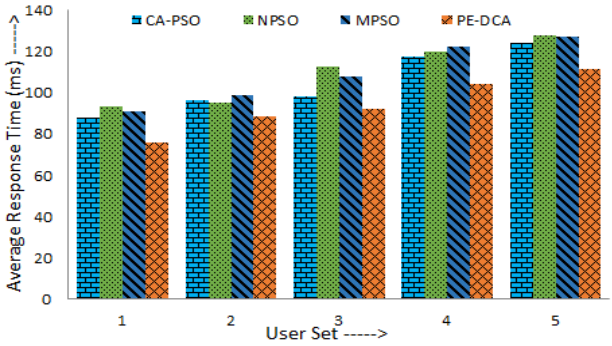


Fig. 15. Comparison of Average Response Time with PSO Based Techniques

9. Related Work

This section elaborates various related work on DC allocation and meta-heuristic search techniques for selecting suitable DC in cloud computing. Firstly, we discuss the recently proposed optimization-based approaches for allocation and then the searching-based techniques for cloud computing are presented.

For assigning the cloud services, the selection of suitable data center is much required. For which different allocation mechanisms have been proposed in the literature. Here, we present only the optimization-based techniques that target to optimize the allocation parameters such as cost and time. Manasrah *et al.* [21] designed a meta-heuristic-based Variable Service Routing Policy (VSBRP) which reduces the data centers overload and significantly minimized the response time and processing time. Jessica *et al.* [26] proposed a Multi-Objective Genetic Algorithm based cloud brokering policy which reduces the response time and cost. The authors simulated the real data over Amazon EC2. Manasrah *et al.* [27] discussed an optimized service broker routing policy based on a differential evolution algorithm. It aimed to minimize response

time, processing time, and overall cost in fog and cloud environments. The authors implemented the new broker policy using six different scenarios.

Pengcheng *et al.*[28] considered the scheduling mechanism as a multi-objective optimization problem and presented the mechanism to optimize the cost and makespan. It eliminates useless resources to narrow down the search space. A fuzzy-based approach to minimize the end-user cost is discussed using the Ant Colony Optimization (ACO) mechanism to allocate the computing and network resources in [29]. Recently, Zhenxin [30] modified Artificial Bee Colony (ABC) to elite-guided ABC for large number of particle problems. Mapetu *et al.*[31] discussed load balancing mechanism using binary PSO to schedule the task with low-cost and low-time complexity. A cost optimization-based mechanism is presented for the enterprise cloud to optimize computing cost, bandwidth cost, and I/O cost for resource allocation [32].

Table 10. Summary of Literature Study

Reference	Optimization Based					Without Penalty Based Sear.	Penalty Based Sear.
	Cost	Response Time	Processing Time	Makespan	Other		
Manasrah *et al.*[21]	×	√	√	×	×	×	×
Jessica *et al.*[26]	√	√	×	×	×	×	×
Manasrah *et al.*[27]	√	√	√	×	×	×	×
Pengcheng *et al.*[28]	√	×	×	√	×	×	×
Mishra *et al.* [29]	√	×	×	√	×	×	×
Mapetu*et al.* [31]	√	×	√	×	×	×	×
Suchintan *et al.*[32]	√	×	×	×	×	×	×
Larumbe *et al.*[34]	×	×	×	×	√	√	×
Tellez *et al.*[35]	×	×	×	×	√	√	×
Pan *et al.*[36]	×	×	×	×	√	√	×
Hamza *et al.*[37]	×	×	×	×	√	√	×
Parida *et al.*[38]	√	√	√	×	×	×	×
Divya *et al.*[39]	√	×	√	×	×	×	×
Ali *et al.*[40]	×	×	×	√	×	√	×
Alkhashai *et al.*[41]	√	×	√	×	×	√	×
Parida *et al.* [42]	√	×	√	×	×	×	×
Proposed PE-DCA	√	√	√	×	×	√	√

Researchers also utilized various searching mechanisms for resource management, task scheduling, VM allocation, and DC allocation in cloud computing. Leonard *et al.*[33] integrated the brokering concept with multi-criteria location-based selection using neighborhood search approaches for virtual machines residing in the multi-cloud era. The author implemented in CloudSim with the Greedy approach and meta-heuristic search. This work also proved to give improved latency and optimized cost. Larumbe *et al.*[34] determined the DC location using the Tabu search optimization technique. The work aims to optimize network performance and CO_2 emission. Tellez *et al.*[35] presented the Tabu search technique for optimal load balancing between cloud and fog nodes. To solve the joint resource allocation task scheduling problem, Pan *et al.*[36]

proposed a Tabu search-based heuristic approach for cloud computing. Hamza *et al.*[37] discussed a hybrid approach using Tabu search and simulated annealing for load balancing in cloud computing. In [38] authors proposed a new meta-heuristic approach for data center allocation with minimum cost, response time, and processing time. The summary of the literature review is presented in Table.10. From the literature study it is found that the penalty estimation and elimination are the major issues while searching for a suitable data center for allocation and is not found from the literature review.

10. Conclusions and Future Work

The datacenter allocation to the on-demand resources in cloud computing is an essential yet open issue for fair allocation with minimized response time, processing, and cost. Due to the dynamic nature of cloud computing, DC allocation is an NP-hard problem. In this work, we study and examine the impact of a penalty during the allocation of resources. While allocating the resource R_i to on-demand request U_i, the penalty may be associated with another user request U_x, which means R_i might be the best suitable resource for U_x. To search the suitable data centers for the on-demand user, we suggest PE-DCA, a new search-based allocation technique addressing the penalty associated with response time, processing time, and cost using the GLS meta-heuristic technique. The selection of data centers for the on-demand resources is established through finding and eliminating the penalties for each user request in allocation to achieve fair allocation. During the implementation, the number of user requests and data centers were configured using the CloudAnalyst tool and the time and cost parameters were computed. The performance was compared and studied with benchmark allocation techniques of CloudAnalyst and other PSO-based techniques (CA-PSO, NPSO, and MPSO). From the simulation results it was found that as compared with the CloudAnalyst benchmark mechanisms, the proposed PE-DCA performs better in minimizing the time and cost parameters as depicted in Figures 11, 12, and 13. The total VM cost of PE-DCA is approximately equal to the evaluated cost of existing techniques of NPSO and MPSO. In contrast, the average response time of PE-DCA is found to be less as compared to NPSO and MPSO. The PE-DCA provides better data center allocation with minimum response time, cost, and processing time.

As future directions, we will further consider additional constraints for data center allocation in cloud computing such as power consumption, deadline constraint workflow, SLA, and QoS. The penalties related to power consumption, task penalty for deadline-based workflow scheduling, evaluation of SLA, and QoS penalty violation will also be investigated in the context of cloud computing. The exploration of other penalty-related searching techniques is also needed to enhance the cloud environment performance. The data center allocation task can be related to energy consumption to formulate an optimized allocation algorithm for cloud systems. The dynamic nature of cloud computing can be further added to explore more on resource allocation mechanism along with multiple objectives. Such approaches can also be cascaded with penalties by eliminating the searching approach to enhance the efficiency of cloud systems.

References

1. Z. Zhang, C. Wu, and D. W. L. Cheung, “A Survey on Cloud Interoperability: Taxonomies, Standards, and Practice,” SIGMETRICS Perform. Eval. Rev., vol. 40, no. 4, pp. 13–22, 2013.
2. Y. Cao, L. Lu, J. Yu, S. Qian, Y. Zhu, and M. Li, “Online cost-rejection rate scheduling for resource requests in hybrid clouds,” Parallel Comput., vol. 81, no. 800, pp. 85–103, 2019.
3. W. Liang, D. Zhang, X. Lei, M. Tang, K. C. Li, and A. Zomaya, “Circuit Copyright Blockchain: Blockchain-based Homomorphic Encryption for IP Circuit Protection,” IEEE Trans. Emerg. Top. Comput., vol. 6750, no. c, pp. 1–11, 2020.
4. A. Shawish and M. Salama, “Cloud Computing: Paradigms and Technologies,” Inter-cooperative Collect. Intell. Tech. Appl., vol. 495, pp. 39–68, 2014.
5. J. L. Sarkar, C. R. Panigrahi, B. Pati, A. K. Saha, and A. Majumder, “MAAS: A mobile cloud assisted architecture for handling emergency situations,” Int. J. Commun. Syst., vol. 33, no. 13, pp. 1–15, 2020.
6. S. C. Nayak, “Multicriteria decision - making techniques for avoiding similar task scheduling conflict in cloud computing,” Int. J. Commun. Syst., no. July 2018, pp. 1–31, 2019.
7. Sasmita Parida, Suvendu Chandan Nayak, et al. “Truthful Resource Allocation Detection Mechanism for Cloud Computing,” in Third International Symposium on Women in Computing and Informatics (WCI ’15), Indu Nair (Ed.). ACM, 2015, pp. 487–491.
8. S. Mohapatra, C. R. Panigrahi, B. Pati, and M. Mishra, “MSA: A task scheduling algorithm for cloud computing,” Int. J. Cloud Comput., vol. 8, no. 3, pp. 283–297, 2019.
9. J. Proaño, C. Carrión, and B. Caminero, “Empirical modeling and simulation of an heterogeneous Cloud computing environment,” Parallel Comput., vol. 83, pp. 118–134, 2019.
10. G. Zou, Z. Qin, S. Deng, K. C. Li, Y. Gan, and B. Zhang, “Towards the optimality of service instance selection in mobile edge computing,” Knowledge-Based Syst., vol. 217, p. 106831, 2021.
11. R. Buyya, S. K. Garg, and R. N. Calheiros, “SLA-Oriented Resource Provisioning for Cloud Computing : Challenges , Architecture , and Solutions,” in International Conference on Cloud and Service Computing, 2011, no. Figure 1, pp. 1–10.
12. J. Li, S. Su, X. Cheng, M. Song, L. Ma, and J. Wang, “Cost-efficient coordinated scheduling for leasing cloud resources on hybrid workloads,” Parallel Comput., vol. 44, pp. 1–17, 2015.
13. S. C. Nayak and C. Tripathy, “Deadline based task scheduling using multi-criteria decision-making in cloud environment,” Ain Shams Eng. J., vol. 9, no. 4, pp. 3315–3324, 2018.
14. S. Nanda, C. R. Panigrahi, and B. Pati, “Emergency management systems using mobile cloud computing: A survey,” Int. J. Commun. Syst., no. May 2019, pp. 1–20, 2020.
15. Parida S., Pati B., Nayak S.C., Panigrahi C.R. (2020) Offer Based Auction Mechanism for Virtual Machine Allocation in Cloud Environment. Proceedings of ICACIE 2018, Volume 2, vol. 2. pp. 339-352, 2020.
16. C. Voudouris, “Chapter 7 Guided Local Search,” Handb. Metaheuristics, pp. 185–218, 2003.
17. A. Alsheddy and E. P. K. Tsang, “Empowerment scheduling for a field workforce,” J. Sched., vol. 14, no. 6, pp. 639–654, 2011.
18. P.H. Mills “Extensions To Guided Local Search: A thesis submitted for the degree of Ph . D . Department of Computer Science University of Essex,” 2002.
19. M. Gendreau and J.-Y. Potvin, Variable Nneighborhood search (chapter), vol. 146. pp.211-238, 2010.
20. B. Wickremasinghe, R. N. Calheiros, and R. Buyya, “CloudAnalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications,” Proc. - Int. Conf. Adv. Inf. Netw. Appl. AINA, pp. 446–452, 2010.

21. A. M. Manasrah, T. Smadi, and A. ALmomani, “A Variable Service Broker Routing Policy for data center selection in cloud analyst,” J. King Saud Univ. - Comput. Inf. Sci., vol. 29, no. 3, pp. 365–377, 2017.
22. J. Huang, R. J. Kauffman, and D. Ma, “Pricing strategy for cloud computing: A damaged services perspective,” Decis. Support Syst., vol. 78, pp. 80–92, 2015.
23. G. Zhao, “Cost-Aware Scheduling Algorithm Based on PSO in Cloud Computing Environment,” Int. J. Grid Distrib. Comput., vol. 7, no. 1, pp. 33–42, 2014.
24. R. Pragaladan and R. Maheswari, “Improve Workflow Scheduling Technique for Novel Particle Swarm Optimization in Cloud Environment,” Int. J. Eng. Res. Gen. Sci., vol. 2, no. 5, pp. 675–680, 2014.
25. S. gaelle Mohamod, Kasim, “Council for Innovative Research,” J. Adv. Chem., vol. 10, no. 1, pp. 2146–2161, 2014.
26. Y. Kessaci, N. Melab, and E. G. Talbi, “A pareto-based genetic algorithm for optimized assignment of VM requests on a cloud brokering environment,” 2013 IEEE Congr. Evol. Comput. CEC 2013, pp. 2496–2503, 2013.
27. A. M. Manasrah and A. B. B. Gupta, “An optimized service broker routing policy based on differential evolution algorithm in fog / cloud environment,” Cluster Comput.,Vol.22, pp. 1639–1653, 2019..
28. P. Han, C. Du, J. Chen, F. Ling, and X. Du, “Cost and makespan scheduling of workflows in clouds using list multiobjective optimization technique,” J. Syst. Archit., Volume 112, pp. 809-837, 2021.
29. S. B. Suchintan Mishra, Arun Kumar Sangaiah,Manmath Narayan Sahoo, “Pareto-optimal cost optimization for large scale cloud systems using joint allocation of resources,” J Ambient Intell Hum. Comput, 2019.
30. Z. Du, D. Han, and K. C. Li, Improving the performance of feature selection and data clustering with novel global search and elite-guided artificial bee colony algorithm, vol. 75, no. 8. Springer US, 2019.
31. J. P. B. Mapetu, Z. Chen, and L. Kong, “Low-time complexity and low-cost binary particle swarm optimization algorithm for task scheduling and load balancing in cloud computing,” Appl. Intell., vol. 49, no. 9, pp. 3308–3330, 2019.
32. S. Mishra, M. N. Sahoo, A. Kumar Sangaiah, and S. Bakshi, “Nature-inspired cost optimisation for enterprise cloud systems using joint allocation of resources,” Enterp. Inf. Syst., no. 0123456789, 2019 (Inpress).
33. L. Heilig, R. Buyya, and S. Voß, “Location-aware brokering for consumers in multi-cloud computing environments,” J. Netw. Comput. Appl., vol. 95, pp. 79–93, 2017.
34. F. Larumbe and B. Sansò, “A tabu search algorithm for the location of data centers and software components in green cloud computing networks,” IEEE Trans. Cloud Comput., vol. 1, no. 1, pp. 22–35, 2013.
35. N. Téllez, M. Jimeno, A. Salazar, and E. D. Nino-Ruiz, “A Tabu search method for load balancing in fog computing,” Int. J. Artif. Intell., vol. 16, no. 2, pp. 106–135, 2018.
36. P. Yi, H. Ding, and B. Ramamurthy, “A Tabu search based heuristic for optimized joint resource allocation and task scheduling in Grid/Clouds,” 2013 IEEE Int. Conf. Adv. Networks Telecommun. Syst. ANTS 2013, 2013.
37. F. Youssef, B. L. El Habib, R. Hamza, Labriji El Houssine, E. Ahmed, and M. Hanoune, “A New Conception of Load Balancing in Cloud Computing Using Tasks Classification Levels,” Int. J. Cloud Appl. Comput., vol. 8, no. 4, pp. 118–133, 2018.
38. S. Parida and B. Pati, “A Cost Efficient Service Broker Policy for Data Center Allocation in IaaS Cloud Model,” Wirel. Pers. Commun., no. 0123456789, 2020 (Inpress).
39. D. Chaudhary and B. Kumar, “A New Balanced Particle Swarm Optimisation for Load Scheduling in Cloud Computing,” J. Inf. Knowl. Manag., vol. 17, no. 1, 2018.
40. A. Al-maamari and F. A. Omara, “Task Scheduling Using PSO Algorithm in Cloud Computing Environments,” Int. J. Grid Distrib. Comput., vol. 8, no. 5, pp. 245–256, 2015.

41. H. M. Alkhashai and F. A. Omara, “An enhanced task scheduling algorithm on cloud computing environment,” Int. J. Grid Distrib. Comput., vol. 9, no. 7, pp. 91–100, 2016.
42. Parida S., Pati B., Nayak S.C., Panigrahi C.R. (2021) Offer Based Auction Mechanism for Virtual Machine Allocation in Cloud Environment. Proceedings of ICACIE 2019, Volume 1, vol. 2. pp. 621-633, 2021.

Sasmita Parida completed B.Tech. and M.Tech. in Computer Science and Engineering from Biju Patnaik University of Technology, India in the year 2004 and 2010 respectively. She is currently working as Assistant Professor in the Department of Computer Science and Technology at GITA Autonomous College, Bhubaneswar, BPUT, India. She has around 15 years of experience in teaching and research. She is pursuing Ph.D. degree at Rama Devi Women′s University, India. Her area of research is cloud computing, Bigdata, IoT, Machine Learning and parallel computing. He has published more than 18 research papers in different journals and conference proceedings.

Bibudhendu Pati completed his Ph.D. degree from IIT Kharagpur, India. He is currently working as Associate Professor in the Department of Computer Science at Rama Devi Women's University, Bhubaneswar, India. He has around 23 years of experience in teaching and research. His areas of research interests include Wireless Sensor Networks, Cloud Computing, Big Data, Internet of Things, and Advanced Network Technologies. He has got several papers published in reputed journals, conference proceedings, and books of international repute. He has been involved in many professional and editorial activities. He is a Life Member of Indian Society for Technical Education, Computer Society of India and Senior Member of IEEE.

Suvendu Chandan Nayak received his Ph.D. degree from Veer Surendra Sai University of Technology (Formally University College of Engineering, Burla) India in Computer Science & Engineering. He is currently working as an Associate Professor in the Department of Computer Science and Information Technology at GITA Autonomous College Bhubaneswar, BPUT, India. He has 14 years of teaching and research experience in the field of computer science. His area of research is cloud computing, Bigdata, IoT and parallel computing. He has published more than 20 papers in different International / National journals and conference proceedings.

Chhabi Rani Panigrahi received her Ph.D. in Computer Science and Engineering from IIT Kharagpur, India. She is currently an Assistant Professor in the Department of Computer Science at Rama Devi Women's University, Bhubaneswar, India. Prior to this, she was working as Assistant Professor in Central University of Rajasthan, India. Her research interests include Software Testing, Mobile Cloud Computing, and Machine Learning. She holds 20 years of teaching and research experience. She has published several international journals, conference papers, and books. She served as chairs and technical program committee member in several conferences of international repute.

Tien-Hsiung Weng is currently working as a Professor in the Department of Computer Science and Information Engineering, Providence University, Taichung City, Taiwan. He received his PhD in Computer Science from the University of Houston, Texas. His research interests include parallel computing, high performance computing, scientific computing, and machine learning. He has has published several international journals, conference papers, and books.

Received: May 12, 2021; Accepted: September 22, 2021.

Computer Science and Information Systems 19(2):709–733 https://doi.org/10.2298/CSIS210518054M

QoS Prediction for Service Selection and Recommendation with a Deep Latent Features Autoencoder

Fatima Zohra Merabet and Djamel Benmerzoug

LIRE Labotory, Faculty of NTIC,
University of Constantine2-Abdelhamid Mehrin Constantine,
Algeria
{$fatima.merabet, djamel.benmerzoug$}@univ-constantine2.dz

Abstract. The number of services on the Internet has increased rapidly in recent years. This makes it increasingly difficult for users to find the right services from a large number of the functionally equivalent candidate. In many cases, the number of services invoked by a user is quite limited, resulting in a large number of missing QoS values and sparseness of data. Consequently, predicting QoS values of the services is important for users to find the exact service among many functionally similar services. However, improving the accuracy of QoS prediction is still a problem. Despite the successful results of the proposed QoS prediction methods, there are still a set of issues that should be addressed, such as Sparsity and Overfitting. To address these issues and improve prediction accuracy. In this paper, we propose a novel framework for predicting QoS values and reduce prediction error. This framework named auto-encoder for neighbor features (*Auto-NF*) consists of three steps. In the first step, we propose an extended similarity computation method based on Euclidean distance to compute the similarity between users and find similar neighbors. In the second step, we form clusters of similar neighbors and partition the initial matrix into sub-matrices based on these clusters to reduce the data sparsity problem. In the third step, we propose a simple neural network autoencoder that can learn deep features and select an ideal number of latent factors to reduce the overfitting phenomenon. To validate and evaluate our method, we conduct a series of experiments use a real QoS dataset with different data densities. The experimental results demonstrate that our method achieves higher prediction accuracy compared to existing methods.

Keywords: similarity computation, neighbors selection, quality of service (QoS), QoS prediction, autoencoder.

1. Introduction

Service recommendation and selection have attracted much attention in the service computing community in recent years [4]. With the dramatic increase in the number of services, different service providers offer many services with the same or similar functions [45]. At the same time, due to the large number of available services, it becomes more difficult for a user to select services that meet his or her requirements [43]. So, it is an urgent task to solve how to recommend suitable services to meet users' requirements. The key criterion considered in recommending services is their Quality of Service (*QoS*), which

can distinguish the suitable services among different functionally equivalent services [45]. Most previous studies have assumed that the quality values of candidate services must be known and accurate. However, it is really hard for a user to invoke all candidate services to acquire their QoS values and make a final decision [45]. Thus, QoS prediction is an indispensable task to finish service selection and recommendation with high quality.

As mentioned above, an active user usually can use only a limited number of services due to the enormous number of them on the Internet what makes the QoS data very sparse (*there are many entries without QoS values*). As a result, the task of QoS prediction to complete the unknown entries in the dataset is really important. Traditional Collaborative Filtering based methods are used to predict missing values. These last based on: 1) Similarity Calculation. Generally, the similarity is calculated using the known QoS values between users and services. And, the widely adopted similarity computational model includes Pearson correlation coefficient, cosine. etc [27]. 2) Neighborhood Selection. In this step, similar neighbors of users and services are identified based on the computed similarities [27]. 3) Collaborative Prediction. Here, the final prediction of QoS values is made by weighting the sum of QoS values of the selected neighbors [27]. Among all prediction methods, collaborative filtering (*CF*) methods have been deeply studied and applied mainly because of their simplicity and effectiveness. However, In QoS prediction, high data sparsity is a common problem and neighborhood-based CF method is not able to learn latent features from historical QoS records. Therefore, the community has proposed a new model-based CF algorithm that attracts more attention to latent features learning. As a typical latent factor model, matrix factorization (*MF*) achieves good performance in learning latent features in high sparsity. Many existing studies extend MF for QoS prediction. However, despite the successful results of MF in the recommendation area, there are still a set of problems that should be handled, as we will mention in the sect. 2. In this study, we aim to address two major issues: 1) the sparsity caused by the service invocation matrix. 2) The overfitting due to the latent factors learning. These two main problems affect prediction accuracy. Therefore, improving the accuracy of QoS prediction has become a challenge.

Recently, deep neural networks have received much attention in many fields and have become increasingly popular [45]. However, few studies use neural network techniques in QoS prediction. One of the most widely used deep learning methods is autoencoder due to the advantages of fast convergence and no labeling requirement. Autoencoder has a simple network structure and also has a strong ability to learn latent features [45].

In this paper, we propose a framework-based autoencoder to alleviate the previous issues and exploit the benefits of improving the quality of neighborhood selection and learning deep latent features from the QoS dataset. The purpose of our model is to minimize the discrepancy between input and output data. The main contributions of our work can be summarized as follows:

- we propose an extended similarity computation method based on Euclidean distance to compute the similarity between users and find similar neighbors. In this method, we use a simple and efficient concept based on common services invoked by both users to improve and facilitate the quality of neighbors' selection;
- we create clusters for similar neighbors according to the computed similarities based on QoS values. And, we partition the initial matrix into small sub-matrices based on these clusters to reduce the data sparsity problem;

- we propose a neural network autoencoder capable of learning deep features and selecting an ideal number of hidden neurons to reduce the overfitting phenomenon in the learning step;
- to validate and evaluate our method, we conduct a series of experiments use a real QoS dataset with different data densities. The experimental results show that our method achieves higher prediction accuracy compared to existing methods.

The remaining sections of this paper are organized as follows. Sect. 2 summarizes related work on QoS prediction. Sect. 3 explains the whole framework. Sect. 4 elaborates an example of motivation. Sect. 5 gives the experimental results. Sect. 6 discusses our results, and sect. 7 concludes the paper.

2. Related Work

To improve the prediction accuracy, many researchers have proposed a series of QoS prediction methods. These can be broadly classified into two categories, i.e., collaborative filtering (*CF*) methods and content-based methods. Most of the proposed methods are extended from CF algorithm [3], [34]. In this section, we introduce a literature review about CF-based QoS prediction method, deep learning-based QoS prediction method, and autoencoder based QoS prediction method.

2.1. CF based QoS Prediction

Collaborative filtering (*CF*) is widely used in web service recommendation and service selection [16] due to its good performance. The basic idea of collaborative filtering is to use historical data for prediction [20][23]. Generally, CF-based QoS prediction methods can be divided into two categories, including neighborhood-based CF and model-based CF [2]. The neighborhood uses the existing QoS values of similar users (*or services*) to predict the missing QoS values, whereas model-based CF approaches build a predefined prediction model trained using historical QoS data to then predict missing QoS values. The improvement of most neighborhood-based CF methods is done by improving or designing new similarity calculation techniques, improving the quality of neighbor selection or combining different methods.

The neighborhood-based CF methods are widely used in QoS prediction and have the advantages of easy implementation and high scalability. Neighborhood-based CF algorithms can be classified into user-based CF methods, item-based CF methods and hybrid CF methods. Nilashi et al. [17] propose a new hybrid recommendation method based on Collaborative Filtering (CF) approaches using dimensionality reduction (SVD) and ontology techniques. Shao et al. [26] proposed a user-based CF with a new user similarity calculation method. Sun et al. [30] proposed a new similarity method for calculating the similarity between two services. Tang et al. [31] proposed a hybrid method to find similar neighbors in users set and services set respectively. Zheng et al. [30] introduced an improved similarity computation method to find similar neighbors.

This category mainly suffers from the data sparsity problem due to the limited number of Web services that a single user invokes, suffers from cold start problems [49][26] i.e., how to recommend a service to a user when there are few or no QoS records.

To alleviate the limitations of neighborhood methods, the community has designed model-based methods that can deal with the data sparsity problem. Papadakis et al. [18] propose a Synthetic Coordinate based Recommendation system. It is parameter-free, so it does not require tuning to achieve high performance and is more resistant to the cold start problem compared to other algorithms. Model-based CF methods use machine learning techniques, such as latent factor models [45], clustering-based models [41], [44], and aspect-based models [28]. Different form neighborhood-based methods that make prediction directly from the ratings of similar neighbors, model-based methods compress user-service rating matrix into a low-dimensional representation in terms of latent features using matrix factorization (*MF*) technique. The most widely applied algorithm that factorizes the rating matrix into two matrices: the user‘s feature matrix and the item's feature matrix, where one row and one column of each matrix are taken as the inner product for prediction [9]. Matrix factorization has emerged as one of the main approaches of model-based method because it can handle sparse matrices and produces good prediction accuracy. It has been verified by many experiments that model-based methods often have better performance than neighborhood-based methods. However, the existing model-based methods can only learn linear relationships between a user and a service, and the inner product cannot catch deep features [6]. Xu et al. [40] introduced a probabilistic matrix factorization model into service recommendation. Chen et al. [5] proposed a fuzzy clustering-based approach to learn latent features of users and services and designed a latent factor model to learn the features of each cluster. Mattiev et al. [15] propose new methods capable of reducing the number of class association rules generated by "classical" classifiers for class association rules, that use distance-based agglomerative hierarchical clustering. Zhu et al. [51] proposed a new context-aware reliability prediction approach, which solves the problem of data sparsity by constructing context-aware reliability models. Wu et al. [36] proposed a deep latent factor model by sequentially connecting multiple latent factor models.

2.2. Deep Learning based QoS Prediction

Recently, deep neural networks have received much attention in many fields and have become increasingly popular. Compared to the pure CF or MF methods, the neural network's method achieve better performance in traditional recommender systems. Some studies have attempted to integrate neural networks into collaborative filtering [1], [10]. However, there are still few studies using neural network techniques in QoS prediction. Jin et al. [8] proposed a deep learning model for predicting QoS of Web service, which builds the model through multi-layer perceptron (*MLP*) and convolution neural networks (*CNN*). Paradarami et al. [19] also presented a deep learning framework to recommend new products to users.

2.3. Autoencoder based QoS prediction

As one of many deep learning methods, autoencoder model is widely applied in recommendation systems for the advantages of fast convergence and no labeling requirement. The auto-encoder has a simple network structure and a strong ability to learn latent features. The autoencoder [24], [7] is an unsupervised neural network and can encode itself

with its latent factors. For example, Liang and Baldwin [13] utilized an autoencoder model to learn the user latent feature matrix for achieving fairly good performance in recommendation. Zhuang et al. [52] proposed a Dual-Autoencoder model to generate latent user and item feature matrices. The existing autoencoder based recommendation systems can be generally divided into two types [46]: one is to learn the latent feature representations of users and items as [21], and the other is to fill the missing value of the original matrix in the reconstruction layer of autoencoder. The authors in [33] tried to use the denoising autoencoder to reduce features dimension. Yin et al. use the autoencoder improved by the substitution strategy to obtain nonlinear latent features of users and services, and missing QoS are generated by the traditional MF methods [45]. Since autoencoder generally uses one hidden-layer neural network to learn the embedding feature, Zhang et al. adopt the MLP to model the nonlinear characteristics of embedding features [47], they also embed similar neighborhoods in MLP to further improve prediction accuracy [8]. Wu et al. proposed a deep neural network for making QoS prediction with contextual information [38], where a deep neural network is added to the end of FM in series for prediction.

The ultimate goal of our work is to solve problems that are complementary to those addressed by the previous works, like data sparsity and overfitting and obtain the deep hidden features by autoencoders. The experimental results presented in sect. 5 demonstrate that Auto-NF significantly outperforms the existing compared methods.

3. The Proposed Framework

Some research works deal with QoS prediction to recommend high-quality services to users in a dynamic environment. However, these works ignore the effect of data sparsity and overfitting and cannot learn deep features. To overcome the major challenges, we propose a high-reliability QoS prediction framework based on an autoencoder neuronal network as shown in (Fig. 1). This framework named autoencoder of neighbor features for short Auto-NF is capable of learning the hidden features to achieve highly accurate QoS prediction.

As shown in (Fig. 1), we assume that the user-service QoS record is the official input of the proposed method. First, we convert this dataset into a matrix with n services in columns and m users in rows. Then, we calculate the similarity between users to obtain neighbors and create sub-matrices with these neighbors. After that, we add features to the matrices of neighbors. Finally, we predict missing QoS values of the services.

3.1. The Framework Procedure

The framework procedure consists of three stages. Stage 1 selects the neighbors of users and services. Stage 2 adds features to the sub-matrices of neighbors. Stage 3 predicts the missing QoS values. The three stages are explained in detail below:

Phase 1 (*neighbors selection*). This step is to identify the similar neighbors of users and services to partitions them into clusters.

1. Similarity calculation: we propose an improved similarity computation method based on common services invoked by both users. This similarity uses historical QoS

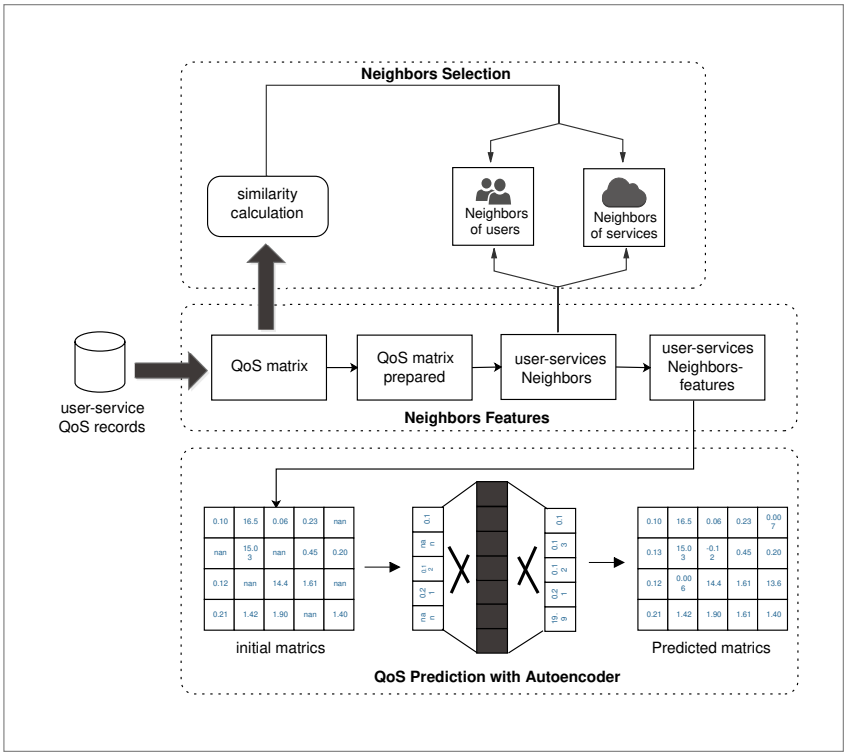


Fig. 1. The proposed framework Auto-NF for QoS prediction

records of services invoked by the same users. And, it can be determined by the two following factors:

- **Common services with different QoS values,** i.e., how many services that users have both invoked in the past;
- **Common services with the same QoS values,** i.e., how similar their QoS values are for services that users have both invoked in the past.

Based on these two factors, we incorporate the following formulate $\lambda_{u,v}$ into the similarity computation method to improve it and then increase the accuracy of neighbor's selection. This formula is computed as:

$$\lambda_{u,v} = \log \frac{|i|}{\left|i_{u,v}\right|} \tag{1}$$

Where $|i|$ is the total number of services, and $\left|i_{u,v}\right|$ represents the number of common services invoked by any two users u and v.
We will now discuss how to consider the two factors defined.[1]

- **First factor.** When two users have both called a large number of services. They have partitioned into the same cluster. And they will likely invoke similar other services;
- **Second factor.** Given a service s invoked by user u and user v, s is considered as a common service for these two users if their properties for service s are similar. If the number of common services is large, they are classified into the same cluster.

The proposed similarity is based on Euclidean distance and computed as follows:

$$S_{u,v} = \frac{1}{1 + \sqrt{\frac{\sum_{i=0}^{M} ((q_{u,i} - \bar{q}_u) - (q_{v,i} - \bar{q}_v))^2 . \lambda_{u,v}}{|M|}}} \tag{2}$$

Where $S_{u,v}$ is the similarity between user u and user v. We form the common behavior service invocation of services co-invoked like: $M = M_u \cap M_v$ which is the set of services invoked by both user u and user v. M_u denotes the set of services invoked by user u and M_v denotes the set of services invoked by user v. $q_{u,i}$ is the real QoS value generated after the target service i is invoked by the target user u, and $q_{v,i}$ is the real QoS value of the target service i is invoked by the target user v. We add a one to the similarity formula to avoid division by zero. $\bar{q}_u$ is the average QoS value of user u, and $\bar{q}_v$ is the average QoS value of user v. The final similarity results are recorded in the clusters of user and service neighbors.

2. Clusters selection: the goal of this step is to partition similar users and services into clusters. To determine reliable clusters, we divide the similarity values into ten intervals of [0, 0.1], [0.1, 0.2], ..., [0.9, 1], and [0, 0]. The users with the same similarity interval are included in the same cluster, and the services invoked by their users are partitioned into

[1] Remark

- Services accessed by the same users are considered as neighbors and placed in the same cluster.
- The services not invoked by any users are not considered;
- The users who have invoked few services and have no shared services with other users are considered untrusted users and not included in any cluster.

the same cluster according to the similarity interval. Each user appears in only one cluster. In this way, we obtain six clusters for users and six clusters for services, i.e., {u1, u6, u8}, {u9}, {u2, u4}, {u3, u5, u7}, {u15,u14}, and {s1, s0, s6}, {s0, s9, s8}, {s1,s2, s4}, {s3, s5, s7},{s0,s2,s20, s13}. The users or services in the same cluster have similar or even the same experiences with certain services. The final clustering results are recorded in the sub-matrices of neighbors.

Phase 2 (*neighbors features*). In the initial matrix, there are n services in columns and m users in rows, where each cell represents a corresponding QoS value assigned to a service by a user. We assume that these QoS contain missing values, resulting in a very sparse matrix. This step aims to reduce this sparsity by preparing the matrix and splitting it into pre-completed sub-matrices of neighbors and features because the denser the matrix, the better the results. To do this, we go through a series of steps described in the sect. 5 of results.

1. Matrices of neighbors: we divide the prepared matrix into sub-matrices that have a different number of rows and columns. The number of rows and columns corresponds to the size of each cluster. thus, each cluster of neighbors is represented with a reduced matrix. Finally, we add features to these sub-matrices and obtain six sub-matrices of neighbors.

2. Matrices of features: simultaneously with the calculation of similarity between users, six features vectors with the same number of clusters are generated. Here each vector represents a particular cluster of neighbors. The size of the vector equals six. As shown in Table 1, a vector initialized with zero receives a one at the specified cell, based on the specified similarity interval for that cluster. After that, the vectors of features and the matrices of neighbors are combined to create the neighbor's features matrices. Then, transmit these matrices to predict their missing QoS values with our autoencoder model.

Table 1. Features of clusters (*users and services*)

Similarity interval	Vectors of features						Clusters of users	Clusters of services
0.2–0.5	1	0	0	0	0	0	Clu_0	Cls_0
0.5–0.6	0	1	0	0	0	0	Clu_1	Cls_1
0.6–0.7	0	0	1	0	0	0	Clu_2	Cls_2
0.7–0.8	0	0	0	1	0	0	Clu_3	Cls_3
0.8–0.9	0	0	0	0	1	0	Clu_4	Cls_4
$0.9-1$	0	0	0	0	0	1	Clu_5	Cls_5

Phase 3 (*QoS Prediction with an Autoencoder*). Autoencoder is an unsupervised model that attempted to reconstruct the input data in the output layer [46] for each training sample through the network. In general, autoencoder can be considered as an extended version of artificial neural network with three or more layers (*an input layer, one or more hidden layers, and an output layer*), where the output layer should have the same size as the input layer. In our work, we propose an improved auto-encoder (an auto-encoder with SBS structure, sect. 3.2) to predict missing QoS values where the numbers of neurons in the

input and output layer are represented by the number of users. The learned autoencoder produces the missing QoS values and shown us the prediction error for these values.

3.2. Structure of Autoencoder

Autoencoder has two types of Structures as shown in (Fig. 2): Small-Big-Small (*SBS*) structure, where the largest layer is in the middle and the smallest layers are at the beginning and end. And Big-Small-Big (*BSB*) structure, where the smallest layer is in the middle and the largest layers are at the beginning and end. We tried both structures on our work and found that using a single hidden layer with a large number of neurons is the best (*SBS*).

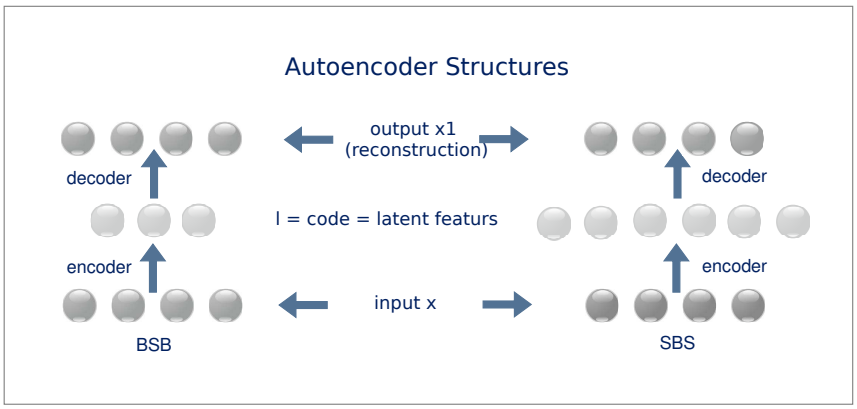


Fig. 2. General structures of autoencoder

4. Motivation and Problem Description

Our method aims to predict missing QoS values with high accuracy so that users can select the optimal service among candidate services. Table 2 shows an example of a matrix with 4 users in columns and 4 services in rows in total. In this matrix, each entry (*e.g.* $q_{1,1}$ *to* $q_{4,4}$) represents a property of the quality value (*e.g. In this paper, we mainly focus on the response time property*). We define the response time as the time duration between a user sending a request and receiving a response of a service (*e.g. Service i1 to Service i4*) observed by a user (*e.g. User u1 to User u4*) in the past. For example, when User u1 invokes Service i1, this response time value is recorded as the first entry $q_{1,1}$. The NaN value in the matrix means that the user has not invoked the service yet. And therefore, there is no record of the quality of service. Since users invoke few services in the real world, the user-service matrix is very sparse. The problem is how to use the known entries in predicting the unknown entries in the user-service matrix. More formally, the problem studied in this paper is defined as follows:

Given a sparse user-service matrix M, the existing entries $val = q_{u,i}$ are used to predict the missing (*NaN*) values. We define the users that invoked the same set of services as S(u) and the services that are commonly invoked by the same set of users as S(i).

Given a user u, a service i, a set of users U = {u1, u2, u3, u4}, and a set of services I = {i1, i2, i3, i4}, the prediction process of the quality of service i mainly consists of three parts: (1) identifying S(u) from U and S(i) from I; where S(u) is the user's neighbors and S(i) is the service's neighbors. (2) Creating the set matrices of features neighbors based on the initial matrix M and the set of neighbors S(u) and S(i). (3) Predict missing QoS values and know the optimal service preferred by user u from the candidate services in matrix M.

Table 2. An example of a user-service matrix

	Service i1	Service i2	Service i3	Service i4
User u1	$q_{1,1}$	$q_{1,2}$	NaN	$q_{1,4}$
User u2	NaN	$q_{2,2}$	$q_{2,3}$	$q_{2,4}$
User u3	$q_{3,1}$	$q_{3,2}$	NaN	$q_{3,4}$
User u4	NaN	NaN	$q_{4,3}$	NaN

From Table 2, we take User u1 as an example. The QoS value of Service i3 is missing for User u1. In this case, we represent it as $q_{1,3}$ and consider User u1 as the target user and Service i3 as the target service. So, u1 only invoke {i1, i2, i4} and we represent its QoS values for these services by $\{q_{1,1}, q_{1,2}, q_{1,4}\}$. While i1 invoked by {u1,u3}, i2 invoked by {u1, u2, u3}, and i4 invoked by {u1, u2, u3}. If we want to predict how user u1 will show the quality of service i3, we need to identify its similar users by calculating the similarities between him and three other users, namely u2, u3, and u4. First of all, we should find the sets of calls for these users. Then, based on these sets we can find the set of neighbors: From the table, we have seen that u2 invoke {i2, i3, i4}, u3 invoke {i1, i2, i4}, and u4 invoke {i3}. We represent their QoS values by $\{q_{2,2}, q_{2,3}, q_{2,4}\}$, $\{q_{3,1}, q_{3,2}, q_{3,4}\}$, and $\{q_{4,3}\}$ respectively. We have seen that u2 and u3 have several commons services with u1. Therefore, we can conclude that by $u1 \cap u3 = \{i1, i2, i4\}$ and $Su1 \cap u2 = \{i2, i4\}$, which means that both users u1 and u3 have used services i1, i2, i4. And both users u1 and u2 have used services i1 and i4, respectively. So, S(u) = u1,u2,u3 and S(i) = i1,i2,i4. In this way, we can obtain $q_{1,3}$, considering the similarity between u1 and u3 as: $\{q_{1,1}, q_{1,2}, q_{1,4}\}$, $\{q_{3,1}, q_{3,2}, q_{3,4}\}$, and between u1 and u2 as: $\{q_{1,2}, q_{1,4}\}$, $\{q_{2,2}, q_{2,4}\}$. In the same way, the remaining missing QoS values can be predicted. Considering u1 and u4, we obtain $S(u1) \cap S(u4) = \{\}$, which means that users u1 and u4 have any services in common. Based on the two factors defined in sect. 3, u1 and u4 are not similar. Therefore, u4 is not helpful and should not be used to predict $q_{1,3}$. Consequently, identifying similar users is of great significance. Meanwhile, it is also important to exclude dissimilar users.

5. Experimental and Evaluation

To evaluate the effectiveness and efficiency of our model, in this section we conducted a set of experiments on a real QoS dataset of Web services. Sect. 5.1 presents the experimen-

tal setup in detail, including parameters settings and dataset description. Sect. 5.2 shows the preparation of the dataset. Sect. 5.3 and sect. 5.4 introduce the methods comparison and the metrics used, respectively. Then, sect. 5.5 analyzes the results, including the evaluation of the table comparison, data validation, prediction, the effect of matrix density, and an example to evaluate our model with another type of dataset called MovieLens. In particular, our experiments aim to answer the following research questions (*RQs*): RQ1: How does the proposed method perform compared to the well-known state-of-the-art QoS prediction methods? Does it provide better prediction accuracy than them? RQ2: Does our model sensitive to over-fitting? RQ4: The data density is usually used to simulate the true scenario in the real world, where the goal is to evaluate the prediction accuracy of the proposed method. So, in our case, what is the performance of the proposed method under different data densities?

5.1. Experimental Setup

Our source code is implemented in python, runs on spyder, and all variants of experiments are performed on a machine with Intel(*R*) Core(*TM*) i7-8550U @ 1.80GHz CPU 1.99GHz with 8Go RAM on a Windows 10 Professional server.

Parameters Settings the parameters used in our approach are shown in Table 3.

Table 3. Important parameters used in our approach

Parameters	**Optimal value**
Regularization	0.001
Learning rate	0.0001
Activation function	Selu
Optimization function	Adam
input and output layers	The number of users in each matrix
Batch size (*hidden layer*)	2048
Layers	[input layer, hidden layer, output layer]

Dataset Description we conducted our experiments using one of the most widely available dataset in the world called WS-DREAM [2]: dataset#2, which was previously collected by Zheng et al [48]. WS-DREAM contains a total of 1,974,675 QoS records obtained from 4500 web services accessed by 142 users from all over the world. There are two attributes for each QoS record in this dataset, response time (*RT*) and throughput (*TP*). However, in this work, we focus on the response time dataset as the evaluation attribute. In total, we have $142 \times 4500 \times 64$ QoS records for each criterion (*response time or throughput*). This dataset has been used in the research community by many researchers [45],[43],[34] to evaluate the accuracy of QoS prediction. The statistics of our experimental dataset are shown in the following Table 4.

[2] https://wsdream.github.io

Table 4. Dataset statistics

Property Value	**RT**	**TP**
Range	0 20s	0 7000kbps
Average	1.33s	11.35kbps
Time slices function	64	64
Time interval	15min	15min
Number of users	142	142
Number of services	4500	4500
Number of all values	30 287 611	30 287 611
Number of missing values	10 609 313	10 609 313
Mean values	3.165s	9.608 kbps

5.2. Dataset Preparation

In real-world service invocation, the number of known QoS records is quite limited. Users typically invoke only a very small number of services, resulting in limited overlap of the same service invocation among users. However, in experiments, a user always invokes thousands of web services. Note that the original dataset is a dense full matrix. To simulate the real scenario and make the data-sparse, we randomly removed some records to obtain a density of 5% to 90%. Then, the sparse data is used as a training set, while the removed QoS value is used as a test set. For example, data with a density of 20% means that 20% of the data is kept as training set and used to build the predictive model, while the removed 80% of the data is used to evaluate the model. This step aims to reduce the number of invalid entries and thus improves the prediction accuracy. In the beginning, our dataset contains missing values. To precompute it, we first need to identify each missing $QoS_{i,u,s,t}$ value in the dataset and then replace it according to the following two rules:

$$QoS_{i,u,s,t} = average \tag{3}$$

$$QoS_{i,u,s,t} = 0 \tag{4}$$

Where $i\epsilon\{response\ time\}$, u is the user ID, s is the service ID, and t is the time slot of the QoS value ($t\epsilon\{0, ..., 63\}$) [29].

We make a comparison between zero and average values as shown in Table 5. We take the smallest and the largest matrix from our set of sub-matrices (matrix 0 and matrix 2) and calculate the error values predicted for both sub-matrices at the highest and lowest densities, and we take the average error values between the two sub-matrices. Where:
Matrix 0: is the smallest matrix in our set of sub-matrices with a size of 17 users in the rows and 4506 services in the cols.
Matrix 2: the largest matrix with a size of 38 users in the rows and 4506 services in the cols.

From Table 5 we can see that zero gives the smallest error of the predicted values. So, the results are better when we replace the missing values with zero than when we replace them with the average values.

In the following, we assume that this prepared dataset will be the official input of our proposed autoencoder model, and now we are ready to start learning it.

Table 5. Comparison between zero and average QoS value based on RMSE and MAE errors

		Training set density —— response time dataset							
		zero				average			
		10%		90%		10%		90%	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Matrix 0	Prediction	0.1121	0.1644	0.0200	0.0287	0.1338	0.1808	0.0236	0.0328
Matrix 2	Prediction	0.1774	0.2416	0.0511	0.0711	0.2044	0.2708	0.0554	0.0761
average	Prediction	0.1447	0.2030	0.0356	0.0499	0.1691	0.2258	0.0395	0.0544

5.3. Comparison

To evaluate the prediction accuracy of our model, we compared it with several well-known QoS prediction models, as listed below.[3] The results are shown in Table 6 and Table 7.

UMEAN (*User Mean*): this method uses the average QoS value known by the active user to predict the missing QoS value of all services invoked by him [45].

IMEAN (*Item Mean*): this method employs the average QoS value on the used services to predict the QoS values of the unused services [45].

UPCC is a user-based Collaborative Filtering method that uses the Pearson Correlation Coefficient to calculate the similarity between users and then predict the missing QoS values based on the historical QoS records of similar users [22].

IPCC is the same as UPCC, except that it calculates the similarity between items [25].

UIPCC is a hybrid method proposed in [14] that combines the advantages of UPCC and IPCC methods to fully exploit the similarity between users and services for predicting missing QoS values.

WSRec this method is a hybrid approach that exploits both similar users and similar services, and linearly combines the prediction results of UPCC and IPCC [48].

PMF (*probabilistic matrix factorization*) is based on probabilistic matrix factorization, which introduces probability models to further improve matrix factorization models, and it also factorizes the user–service QoS matrix for the prediction [16].

NMF (*Non-negative Matrix Factorization*) this method applies non-negative matrix factorization to the user-item matrix to predict missing values without considering neighborhood information [11].

CAP (*Credibility-Aware Prediction*) CAP is a novel credibility-aware QoS prediction method that uses two-phase K-means clustering algorithms [35].

CMF (*Classic Matrix Factorization*) CMF is the classical matrix factorization method whose main objective is to builds a global model to make quality predictions based on the available quality information [9].

LFM (*Latent Factor Model*) LFM decomposes the user-service QoS matrix into a low-dimensional reduction to learn latent features of users and services and then predicts results based on the learned latent features [9].

SN-MF (*Service Neighbors-based MF*). SN-MF proposes three service-neighborhood enhanced prediction models for selecting neighbors of each service with the integration of latent features of service neighbors into the basic matrix factorization model [42].

[3] All parameters in the widely-used models are set to the same values as in the original papers.

LE-MF (*Linear-Ensemble MF*). LE-MF integrates context information of users and services respectively and trusts mechanism into traditional matrix factorization model to predict QoS values [39].

LR-MF (*Location and Reputation-aware MF*).LR-MF combines both the user's reputation and location information into the matrix factorization (MF) model to predict the missing QoS values [12].

NIMF (*Neighborhood-Integrated Matrix Factorization*). This method was the first one that integrates neighborhood-based information of users into the MF-based model to achieve higher quality predictions. It computes the PCCs to identify N(u) [50].

NAMF. This method also integrates users' neighborhood information to make quality predictions. Unlike NIMF, it identifies N(u) based on their geographical locations and the network map used to measure the network distance between users [32].

CSMF CSMF is a general context-sensitive matrix factorization method to make collaborative QoS predictions. By considering the complexity of service invocations, CSMF proposes to model user-to-service and environment-to-environment interactions simultaneously and fully exploit implicit and explicit contextual factors in QoS data [37].

JCM (*Joint CNN-MF*) is a new matrix factorization (MF) model which integrates a convolutional neural network (CNN). JCM is capable of learning deep latent features of a user or a service neighbor. JCM incorporates a novel similarity computation method to improve the accuracy of neighbor selection in edge computing environments.[43].

Table 6. Prediction accuracy comparison with different prediction methods in low data densities (*A smaller value means a better performance*)

Model	Training set density —— response time dataset							
	d = 5%		d = 10%		d = 15%		d = 20%	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
UserMean	0.871	1.858	0.873	1.856	0.873	1.856	0.879	1.853
ItemMean	0.742	1.577	0.728	1.548	0.711	1.530	0.700	1.530
UPCC	0.955	2.126	0.782	1.856	0.671	1.726	0.597	1.717
IPCC	1. 102	2.258	0.878	1.989	0.784	1.862	0.722	1.794
UIPCC	0.847	1.920	0.729	1.730	0.612	1.590	0.552	1.587
CMF	0.611	1.414	0.516	1.356	0.491	1.216	0.459	1.198
NMF	0.618	1.574	0.604	1.549	0.599	1.534	0.598	1.533
PMF	0.567	1.473	0.499	1.286	0.472	1.216	0.449	1.182
NIMF	0.551	1.407	0.485	1.274	0.453	1.198	0.435	1.167
NAMF	0.538	1.385	0.485	1.259	0.452	1.207	0.435	1.144
CNMF	0.528	1.305	0.471	1.237	0.431	1.136	0.413	1.116
CAP	/	/	0.360	0.643	/	/	0.352	0.664
WSRec	0.679	1.488	0.621	1.426	0.603	1.368	0.602	1.351
LFM	0.578	1.501	0.564	1.320	0.543	1.271	0.535	1.226
SN-MF	0.631	1.396	0.537	1.269	0.500	1.226	0.487	1.207
LE-MF	0.573	1.370	0.513	1.251	0.482	1.204	0.464	1.180
LR-MF	0.551	1.415	0.478	1.257	0.446	1.212	0.434	1.142
JCM	0.513	1.332	0.466	1.250	0.450	1.185	0.436	1.180
Our	**0.292**	**0.399**	**0.153**	**0.214**	**0.120**	**0.169**	**0.101**	**0.142**

Table 7. Prediction accuracy comparison with different prediction methods in high data densities (*A smaller value means a better performance*)

Model	Training set density —— response time dataset							
	d = 30%		d = 50%		d = 70%		d = 90%	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
UserMean	0.868	1.835	0.877	1.859	0.877	1.866	0.872	1.841
ItemMean	0.682	1.529	0.674	1.504	0.674	1.518	0.680	1.526
UPCC	0.586	1.502	0.607	1.620	0.562	1.473	0.563	1.379
IPCC	0.631	1.472	0.586	1.618	0.543	1.470	0.668	1.417
CAP	0.331	0.678	/	/	0.228	0.581	0.188	0.582
WSRec	0.480	1.342	0.465	1.271	0.431	1.205	0.416	1.132
LFM	0.476	1.351	0.421	1.250	0.401	1.151	0.384	1.101
JCM	0.429	1.174	0.406	1.148	0.389	1.115	0.378	1.089
Our	**0.077**	**0.108**	**0.054**	**0.075**	**0.043**	**0.057**	**0.038**	**0.054**

5.4. Metrics

We evaluated the prediction accuracy of our model compared to other methods using the following metrics, including standard error metrics such as mean absolute error (*MAE*) and root mean square error (*RMSE*).

Mean Absolute Error (*MAE*): MAE is a quantity used to measure the prediction accuracy of QoS prediction methods; it indicates the average of the absolute difference between the predicted QoS value and the real QoS value of the service invoked by a user over the test records.

MAE is defined as follows.

$$MAE = \frac{\sum_{(u,i)\varepsilon M_t} \left| q_{u,i} - \hat{q}_{u,i} \right|}{\left| N(M_t) \right|} \tag{5}$$

Root Mean Square Error (*RMSE*): During the calculation of RMSE, the individual differences between the predicted values and the corresponding observed values are each squared and then averaged over the sample. Then the square root of the average is taken as the final result.

RMSE is defined as follows.

$$RMSE = \sqrt{\frac{\sum_{(u,i)\varepsilon M_t} (q_{u,i} - \hat{q}_{u,i})^2}{\left| N(M_t) \right|}} \tag{6}$$

Where M_t denotes the test set matrix and $N(M_t)$ is the number of QoS values in the test set M_t , $q_{u,i}$ and $q_{u,i}$ represent the real QoS value and the predicted QoS value, respectively. According to the above two definitions, MAE and RMSE both vary in (*0,* ∞), and a smaller value of them means higher prediction accuracy. They are widely used in the field of web service prediction.

5.5. Analyze of Our Results

In the following sections, we conduct a series of experiments to evaluate our model on different training sets with different densities (*5%, 10%, 15%, 20%, 30%, 50%, 70%, 90%*) and we investigate how the density key affects the prediction accuracy. The experiments are conducted on the response time dataset.

Evaluation of the Tables of Comparison From Table 6 and Table 7 and in comparison with the state-of-the-art and other methods, we can make the following observations:

- Our Auto-NF model achieves the lowest errors for both MAE and RMSE metrics in all cases of data densities, including low and high data densities. This indicates that our method can be applied to both sparse and dense datasets;
- Our Auto-NF model performs better prediction accuracy than all other prediction methods compared;
- Our Auto-NF model can better handle the data sparsity problem.

Impact of matrix density Fig. 3 shows the influence of density on prediction accuracy using MAE and RMSE errors. This figure shows that as the data density increases, both MAE and RMSE values decrease rapidly at first. However, they are still the same after the density increases. Therefore, our model produces the lowest prediction errors and has the best results in high density. The reason is that the more the data is trained, the better the quality of learning deep features becomes.

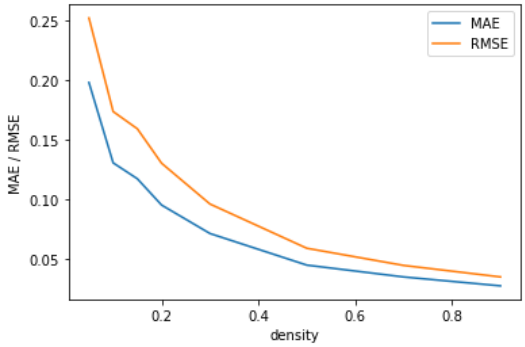


Fig. 3. The MAE and RMSE variation under different densities

Data validation To check if our model is sensitive to the overfitting problem, we needed to follow the evolution of the training and validation graphs based on MAE and RMSE metrics in the learning step and then compared their changes over time.
From Fig. 4 and Fig. 5, we can see that the gap between MAE and RMSE in the training and validation graphs is small, and when we attend a high density, the two graphs coincide, which means that our model is not overfitted.

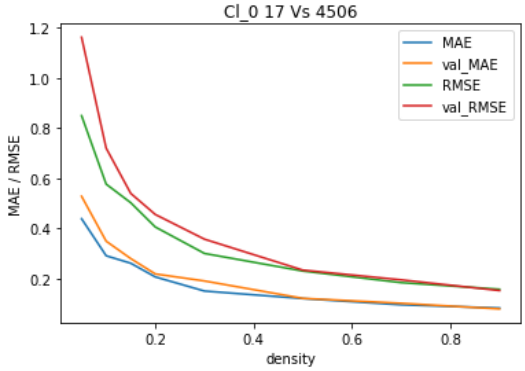


Fig. 4. The evolution of training and validation data based on MAE and RMSE errors in matrix 0

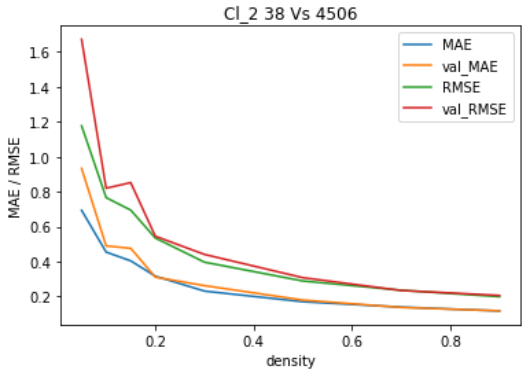


Fig. 5. The evolution of training and validation data based on MAE and RMSE errors in matrix 2

Prediction After the training step, first, we took the test set data from both the smallest and largest sub-matrices from our set of neighbor matrices as missing values and predicted their values with our autoencoder model. Then, we compared original and predicted values to show the difference between them and then evaluated the performance of our model.

For more explanation, we used MAE and RMSE measures to evaluate the predicted values compared to the real ones (*test set values*) and stored the difference between them as (*MAE, RMSE*) prediction error, as shown in Table 8 below.

Table 8. MAE and RMSE errors between some random real and predicted values for the evaluated model

Real value	7.489	1.519	2.570	3.768	1.389
Predicted value	6.991	1.433	2.255	3.570	1.024
MAE prediction error	0.26780975				
RMSE prediction error	0.3974242				
Real value	0	0.002	6.407	0	0.533
Predicted value	0.0009	-0.072	6.457	-0.037	0.548
MAE prediction error	0.04516501				
RMSE prediction error	0.05946906				

From Table 8, it can be seen that the predicted values are almost the same as the real ones in both sub-matrices (first, second, fifth, and sixth rows). There are small values for MAE and RMSE errors, especially when the actual and predicted values are close to zero (third, fourth, seventh, and eighth rows). This assures the performance and the high accuracy of our proposed model.

Real value is represented as:

$$realvalue = \int (predicted\, value) + (Error\, term\, predicton) \tag{7}$$

Where $\int$ is the activation function, the error term is MAE or RMSE used in this work.

Example MovieLens dataset [4]: is one of the well-known movie datasets that has been used for the evaluation of recommender systems. The numbers of users and movies in the Movielens dataset are 69878 and 10677, respectively. In this dataset, the users have provided ratings on a 5-star scale. Hence, based on the number of users and movies, this dataset includes 100, 000, 54 anonymous ratings.
For this example, we have selected 500 users and 5896 movies from the dataset Movielens.
We proceeded in the same way as for the WS-Dream dataset. We have got two clusters for neighbors for both (users and services). Each cluster is represented by a small matrix. Finally, we have obtained two sub-matrix of neighbors features.
Matrix 0: with a size of 9 users in the rows and 3069 movies in the cols.

[4] http://www.movieLens.org

Matrix 1:with a size of 495 users in the rows and 615 movies in the cols.
First, we replaced the missing QoS values in the sub-matrices once with zero and once with the average values. Then, we calculated the error values predicted for both sub-matrices at the highest and lowest densities. Finally, we compared between zero and average values, as shown in Table 9 follow:

Table 9. Comparison between zero and average QoS value based on RMSE and MAE errors (MovieLens dataset)

		Training set density —— MovieLens dataset							
		zero				average			
		10%		90%		10%		90%	
		MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Matrix 0	Prediction	0.0101	0.0154	0.0059	0.0078	0.0456	0.0640	0.0099	0.0142
Matrix 1	Prediction	0.4748	0.7549	0.3564	0.6092	0.2456	0.3917	0.2006	0.3472
average	Prediction	0.2424	0.3851	0.1811	0.3085	0.1456	0.2278	0.1052	0.1807

From Table 9 we can see that average gives the small error of the predicted values. So, the results are better when we replace the missing values with average than when we replace them with the zero values.

To make a new evaluation of our model, we compared the results of its last experiments on the movielens dataset with the previous ones on the WSDREAM dataset. The results are shown in Table 10 and Table 11.

Table 10. Prediction accuracy of our method compared between WSDREAM and MovieLens at low data density (*A smaller value means a better performance*)

Model	Training set density —— WSDREAM Vs MovieLens datasets							
	d = 5%		d = 10%		d = 15%		d = 20%	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Auto-NF WSDREAM	**0.292**	**0.399**	**0.153**	**0.214**	**0.120**	**0.169**	**0.101**	**0.142**
Auto-NF MovieLens	**0.153**	**0.236**	**0.144**	**0.227**	**0.140**	**0.222**	**0.137**	**0.218**

Table 11. Prediction accuracy of our method compared between WSDREAM and MovieLens at high data densities (*A smaller value means a better performance*)

Model	Training set density —— WSDREAM Vs MovieLens dataset							
	d = 30%		d = 50%		d = 70%		d = 90%	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
Auto-NF WSDREAM	**0.077**	**0.108**	**0.054**	**0.075**	**0.043**	**0.057**	**0.038**	**0.054**
Auto-NF MovieLens	**0.125**	**0.203**	**0.116**	**0.191**	**0.108**	**0.181**	**0.107**	**0.183**

Table 10 and Table 11 show that our model evaluated with movielens gives good results. However, most values of the WSDREAM dataset are better than that of movielens (the errors values for both MAE and RMSE in WSDREAM are smaller than those in the movielens). As a result, movieLens provides lower prediction accuracy than the WSDREAM of our model, although it performs well.

6. Discussion

Our principal goal is to select optimum services and recommend them to users based on their Quality of Service, which is the most important criterion considered in recommending services. However, due to the large number of available services on the internet, it is really hard for a user to invoke all candidate services to acquire their QoS values and then make a final decision about the optimal one. Thus, predicting the QoS values of services is an indispensable task to finish service selection and recommendation.
In summary, in this study, we have proposed a model to predict the missing QoS values of services using WSDREAM, the most important dataset in the domain of web services. The experimental results in section. 5 confirm that our method has improved prediction accuracy and reduced susceptibility to overfitting.
Using the WSDREAM dataset of web services allowed us to train our predictor model correctly and with a good performance. However, using a different dataset could lead to lower accuracy. To verify that, we have tried our model with MovieLens, the well-known dataset in the field of movie recommendation see **"Example"** in the previous section. 5.5

Table 12 aims to present the difference between our work and the works proposed in the papers of Yin et al. (2019) [43] and Smahi et al. (2018) [29].
From the experimental results section. 5 and Table 12, we derive the following findings:

Users in the same country may not be neighbors. They may not call the same services, or they may have different QoS values for the same services. As shown in the Table 13 below:

From Table 13, we can observe that the QoS value (the response time) of u1 invoking s1 is very close to u29, even though they are not in the same country. Similarly, u0 and u2 from different countries intend to select the same service s5.
Since service s0 and service s1 are in the same network, their QoS values largely depend on the network distance among users and services.

6.1. Advantages of our model

- Has a simple structure (autoencoder with input-output and one hidden layer);
- Capable of reading deeply hidden features;
- Handle the data sparsity problem;
- Less sensitive to the overfitting problem;
- Achieves the lowest errors for both MAE and RMSE in all cases of data densities, including low and high data densities;
- Outperforms all previous methods in comparison (see tables 6 and 7);
- Performed better prediction accuracy;
- Our model has succeeded in minimizing the discrepancy between input and output data (see Table 8).

Table 12. Comparison between our work and the works presented in the papers of: Yin et al. (2019) [43] and Smahi et al. (2018) [29]

	Our work	Yin et al. (2019) [43]	Smahi et al. (2018) [29]
Similarity computation	We propose a simple new concept, i.e., $\lambda_{u,v}$ based on the common services invoked by both users. $\lambda_{u,v}$ is integrated into Euclidean distance to compute the similarity between two users.	Yin, Y et al propose two new concepts, i.e., IIFU (inverse invocation frequency of users) and IIFS (inverse invocation frequency of services). IIFU and IIFS are integrated into Euclidean distance to compute the similarity between two users or two services.	/
Neighbors selection based clustering	We divide the input dataset into a series of clusters to reduce the data sparsity based on the QoS values. Each cluster is represented with a reduced matrix that has fewer columns and rows concerning the initial dataset.	/	Smahi, M.I. et al divide the input dataset into a series of clusters to reduce the data sparsity based on the country ID or the provider ID. Each cluster is represented with a reduced matrix that has fewer columns and rows concerning the initial dataset.
Data density values	5%, 10%, 15%, 20%, 30%, 50%, 70%, 90%	5%, 10%, 15%, 20%, 30%, 50%, 70%, 90%	80%
Over-fitting	In the learning step, we try with different sizes of hidden layer. Finally we find it 2048 neurons.	/	In the learning step, Smahi, et al perform a cross-validation to infer the best hidden layer size. Finally, they find it 120 neurons.
MAE and RMSE errors	Our model achieves the lowest errors for both MAE and RMSE in all cases of data densities, including low and high data densities. See tables 6 and 7 ”our” model.	See tables 6 and 7 ”JCM” model.	For 80% density and 120 neurons: MAE: 0.681 RMSE:1.369 See original paper.
Dataset	WS-DREAM Movielens ml10M	WS-DREAM	WS-DREAM
Technique	Autoencoder	CNN + MF	Autoencoder

Table 13. A capture from the real-world service invocation scenario rtdata from wsdream dataset

	S0 United State	S1 United State	S5 United State	S14 Argentina	S50 Australia
U0 United State	4.18	0.416	20.0	1.469	nan
U1 United State	1.166	0.335	20.0	0.653	0.771
U2 Japan	5.975	0.251	20.0	0.581	0.823
U29 United Kingdom	1.997	0.324	20.0	0.646	0.888
U30 Canada	1.638	2.408	20.0	10.755	4.132

6.2. Point limits of our model

– Using a different type of dataset like MovieLens could lead to lower accuracy.

7. Conclusion and Future Work

Due to the increasing number of services on the internet, it becomes harder to find the right ones. Furthermore, it is impracticable to check all services for their quality values since this consumes many resources. Therefore, users invoke a quite limited number of services, resulting in a sparse amount of data. Thus, a QoS prediction method is very important to find the most appropriate service among many functionally similar ones. In this work, we propose an effective model for predicting missing QoS values of services. We use historical QoS records for this purpose. Here we split the original dataset into partial matrices to eliminate sparsity and learn deep latent features to reduce the overfitting problem. Experimental results on a public dataset demonstrated that our model achieved the best results compared to the traditional and counterpart methods, both in low and high data densities. In contrast to other deep learning-based recommendation methods, that can achieve better results only in high data density. In the future, we plan to elaborate on another based-deep learning model, such as a convolutional neural network. Also, we will continue to improve our model.

Acknowledgments. The authors would like to acknowledge the ComSIS editor and reviewers for the thoughtful reading of this manuscript and for their relevant comments that helped us improve the quality of the paper.

References

1. Batmaz, Z., Yurekli, A., Bilge, A., Kaleli, C.: A review on deep learning for recommender systems: challenges and remedies. Artificial Intelligence Review 52(1), 1–37 (2019)
2. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. arXiv preprint arXiv:1301.7363 (2013)
3. Chen, L., Ha, W.: Reliability prediction and qos selection for web service composition. International Journal of Computational Science and Engineering 16(2), 202–211 (2018)

4. Chen, S., Fan, Y., Tan, W., Zhang, J., Bai, B., Gao, Z.: Service recommendation based on separated time-aware collaborative poisson factorization. J. Web Eng. 16(7&8), 595–618 (2017)
5. Chen, S., Peng, Y., Mi, H., Wang, C., Huang, Z.: A cluster feature based approach for qos prediction in web service recommendation. In: 2018 IEEE Symposium on Service-Oriented System Engineering (SOSE). pp. 246–251. IEEE (2018)
6. He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S.: Neural collaborative filtering. In: Proceedings of the 26th international conference on world wide web. pp. 173–182 (2017)
7. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. science 313(5786), 504–507 (2006)
8. Jin, Y., Wang, K., Zhang, Y., Yan, Y.: Neighborhood-aware web service quality prediction using deep learning. EURASIP Journal on Wireless Communications and Networking 2019(1), 1–10 (2019)
9. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer 42(8), 30–37 (2009)
10. Kuang, L., Gong, T., OuYang, S., Gao, H., Deng, S.: Offloading decision methods for multiple users with structured tasks in edge computing for smart cities. Future Generation Computer Systems 105, 717–729 (2020)
11. Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. Nature 401(6755), 788–791 (1999)
12. Li, S., Wen, J., Luo, F., Cheng, T., Xiong, Q.: A location and reputation aware matrix factorization approach for personalized quality of service prediction. In: 2017 IEEE International Conference on Web Services (ICWS). pp. 652–659. IEEE (2017)
13. Liang, H., Baldwin, T.: A probabilistic rating auto-encoder for personalized recommender systems. In: Proceedings of the 24th ACM International on Conference on Information and Knowledge Management. pp. 1863–1866 (2015)
14. Ma, H., King, I., Lyu, M.R.: Effective missing data prediction for collaborative filtering. In: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 39–46 (2007)
15. Mattiev, J., Kavšek, B.: Distance based clustering of class association rules to build a compact, accurate and descriptive classifier. Computer Science and Information Systems (00), 37–37 (2020)
16. Mnih, A., Salakhutdinov, R.R.: Probabilistic matrix factorization. Advances in neural information processing systems 20, 1257–1264 (2007)
17. Nilashi, M., Ibrahim, O., Bagherifard, K.: A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques. Expert Systems with Applications 92, 507–520 (2018)
18. Papadakis, H., Panagiotakis, C., Fragopoulou, P.: Scor: a synthetic coordinate based recommender system. Expert Systems with Applications 79, 8–19 (2017)
19. Paradarami, T.K., Bastian, N.D., Wightman, J.L.: A hybrid recommender system using artificial neural networks. Expert Systems with Applications 83, 300–313 (2017)
20. Radovanović, S., Delibašić, B., Suknović, M.: Predicting dropout in online learning environments. Computer Science and Information Systems (00), 53–53 (2020)
21. Rama, K., Kumar, P., Bhasker, B.: Deep autoencoders for feature learning with embeddings for recommendations: a novel recommender system solution. Neural Computing and Applications pp. 1–11 (2021)
22. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: Grouplens: An open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 ACM conference on Computer supported cooperative work. pp. 175–186 (1994)
23. Rogić, S., Kašćelan, L.: Class balancing in customer segments classification using support vector machine rule extraction and ensemble learning. Computer Science and Information Systems (00), 52–52 (2020)

24. Rumelhart, D.E., McClelland, J.L., Group, P.R., et al.: Parallel distributed processing, vol. 1. IEEE Massachusetts (1988)
25. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th international conference on World Wide Web. pp. 285–295 (2001)
26. Shao, L., Zhang, J., Wei, Y., Zhao, J., Xie, B., Mei, H.: Personalized qos prediction forweb services via collaborative filtering. In: Ieee international conference on web services (icws 2007). pp. 439–446. IEEE (2007)
27. Shen, L., Pan, M., Liu, L., You, D., Li, F., Chen, Z.: Contexts enhance accuracy: On modeling context aware deep factorization machine for web api qos prediction. IEEE Access 8, 165551–165569 (2020)
28. Singla, P., Richardson, M.: Yes, there is a correlation: -from social networks to personal behavior on the web. In: Proceedings of the 17th international conference on World Wide Web. pp. 655–664 (2008)
29. Smahi, M.I., Hadjila, F., Tibermacine, C., Merzoug, M., Benamar, A.: An encoder-decoder architecture for the prediction of web service qos. In: European conference on service-oriented and cloud computing. pp. 74–89. Springer (2018)
30. Sun, H., Zheng, Z., Chen, J., Lyu, M.R.: Personalized web service recommendation via normal recovery collaborative filtering. IEEE Transactions on Services Computing 6(4), 573–579 (2012)
31. Tang, M., Jiang, Y., Liu, J., Liu, X.: Location-aware collaborative filtering for qos-based service recommendation. In: 2012 IEEE 19th international conference on web services. pp. 202–209. IEEE (2012)
32. Tang, M., Zheng, Z., Kang, G., Liu, J., Yang, Y., Zhang, T.: Collaborative web service quality prediction via exploiting matrix factorization and network map. IEEE Transactions on Network and Service Management 13(1), 126–137 (2016)
33. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A., Bottou, L.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. Journal of machine learning research 11(12) (2010)
34. Wang, S., Zhao, Y., Huang, L., Xu, J., Hsu, C.H.: Qos prediction for service recommendations in mobile edge computing. Journal of Parallel and Distributed Computing 127, 134–144 (2019)
35. Wu, C., Qiu, W., Zheng, Z., Wang, X., Yang, X.: Qos prediction of web services based on two-phase k-means clustering. In: 2015 ieee international conference on web services. pp. 161–168. IEEE (2015)
36. Wu, D., Luo, X., Shang, M., He, Y., Wang, G., Zhou, M.: A deep latent factor model for high-dimensional and sparse matrices in recommender systems. IEEE Transactions on Systems, Man, and Cybernetics: Systems (2019)
37. Wu, H., Yue, K., Li, B., Zhang, B., Hsu, C.H.: Collaborative qos prediction with context-sensitive matrix factorization. Future Generation Computer Systems 82, 669–678 (2018)
38. Wu, H., Zhang, Z., Luo, J., Yue, K., Hsu, C.H.: Multiple attributes qos prediction via deep neural model with contexts. IEEE Transactions on Services Computing (2018)
39. Xu, Y., Yin, J., Deng, S., Xiong, N.N., Huang, J.: Context-aware qos prediction for web service recommendation and selection. Expert Systems with Applications 53, 75–86 (2016)
40. Xu, Y., Yin, J., Lo, W., Wu, Z.: Personalized location-aware qos prediction for web services using probabilistic matrix factorization. In: International Conference on Web Information Systems Engineering. pp. 229–242. Springer (2013)
41. Xue, G.R., Lin, C., Yang, Q., Xi, W., Zeng, H.J., Yu, Y., Chen, Z.: Scalable collaborative filtering using cluster-based smoothing. In: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 114–121 (2005)
42. Yin, J., Xu, Y.: Personalised qos–based web service recommendation with service neighbourhood–enhanced matrix factorisation. International Journal of Web and Grid Services 11(1), 39–56 (2015)

43. Yin, Y., Chen, L., Xu, Y., Wan, J., Zhang, H., Mai, Z.: Qos prediction for service recommendation with deep feature learning in edge computing environment. Mobile Networks and Applications pp. 1–11 (2019)
44. Yin, Y., Xu, Y., Xu, W., Gao, M., Yu, L., Pei, Y.: Collaborative service selection via ensemble learning in mixed mobile network environments. Entropy 19(7), 358 (2017)
45. Yin, Y., Zhang, W., Xu, Y., Zhang, H., Mai, Z., Yu, L.: Qos prediction for mobile edge service recommendation with auto-encoder. IEEE Access 7, 62312–62324 (2019)
46. Zhang, S., Yao, L., Sun, A., Tay, Y.: Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR) 52(1), 1–38 (2019)
47. Zhang, Y., Yin, C., Wu, Q., He, Q., Zhu, H.: Location-aware deep collaborative filtering for service recommendation. IEEE Transactions on Systems, Man, and Cybernetics: Systems (2019)
48. Zheng, Z., Ma, H., Lyu, M.R., King, I.: Wsrec: A collaborative filtering based web service recommender system. In: 2009 IEEE International Conference on Web Services. pp. 437–444. IEEE (2009)
49. Zheng, Z., Ma, H., Lyu, M.R., King, I.: Qos-aware web service recommendation by collaborative filtering. IEEE Transactions on services computing (2010)
50. Zheng, Z., Ma, H., Lyu, M.R., King, I.: Collaborative web service qos prediction via neighborhood integrated matrix factorization. IEEE Transactions on Services Computing 6(3), 289–299 (2012)
51. Zhu, J., He, P., Xie, Q., Zheng, Z., Lyu, M.R.: Carp: context-aware reliability prediction of black-box web services. In: 2017 IEEE International Conference on Web Services (ICWS). pp. 17–24. IEEE (2017)
52. Zhuang, F., Zhang, Z., Qian, M., Shi, C., Xie, X., He, Q.: Representation learning via dual-autoencoder for recommendation. Neural Networks 90, 83–89 (2017)

Fatima Zohra Merabet is a Ph.D. student at faculty of new information and communication technologies, University of Constantine 2 - Abdelhamid Mehri and affiliated to the LIRE laboratory. He received a license and Master's degrees in Information Systems and web technology in respectively 2016 and 2018 from the same faculty. His research interests revolve around: self-adaptive system, autoencoder, IoT, QoS prediction.

Djamel Benmerzoug is currently a full professor in the department of TLSI, Faculty of New Technologies of Information and Communication, University Constantine 2, Algeria. He holds a PhD in Computer science from Pierre & Marie Curie University (Paris - France). Pr Djamel Benmerzoug has published many articles in many International Conferences and Journals. He supervises many PhD and Master students. His current research interests include Internet of Things, Cloud Computing, Advanced Enterprises Systems, Multiagent Systems, Service Oriented Computing, and Business Processes Modelling and Verification.

Received: May 18, 2021; Accepted: October 05, 2021.

Computer Science and Information Systems 19(2):735–762 https://doi.org/10.2298/CSIS210606003M

ProRes: Proactive Re-Selection of Materialized Views

Mustapha Chaba Mouna[1], Ladjel Bellatreche[2], and Narhimene Boustia[1]

[1] LRDSI Laboratory, Faculty of Science, University Blida 1
Blida, Algeria {mustapha.medea, nboustia}@gmail.com
[2] LIAS/ISAE-ENSMA, Poitiers, France
bellatreche@ensma.fr

Abstract. Materialized View Selection is one of the most studied problems in the database field, covering SQL and NoSQL technologies as well as different deployment infrastructures (centralized, parallel, cloud). This problem has become more complex with the arrival of data warehouses, being coupled with the physical design phase that aims at optimizing query performance. Selecting the best set of materialized views to optimize query performance is a challenging task. Given their importance and the complexity of their selection, several research efforts both from academia and industry have been conducted. Results are promising – some solutions are being implemented by commercial and open-source DBMSs –, but they do not factor in the following properties of nowadays analytical queries: **(i)** large-scale queries, **(ii)** their dynamicity, and **(iii)** their high interaction. Studies to date fail to consider that complete set of properties. Considering the three properties simultaneously is crucial regarding today's analytical requirements, which involve dynamic and interactive queries. In this paper, we first present a concise state of the art of the materialized view selection problem (VSP) by analyzing its ecosystem. Secondly, we propose a proactive re-selection approach that considers the three properties concurrently. It features two main phases: offline and online. In the offline phase, we manage a set of the first queries based on a given threshold δ by selecting materialized views through a hypergraph structure. The second phase manages the addition of new queries by scheduling them, updates the structure of the hypergraph, and selects new views by eliminating the least beneficial ones. Finally, extensive experiments are conducted using the Star Schema Benchmark data set to evaluate the effectiveness and efficiency of our approach.

Keywords: Materialized Views, Hypergraphs, Query Sharing, large-scale of queries.

1. Introduction

Selection of materialized views is a challenging task for designing advanced database applications such as Analytical Databases [24], Autonomous Databases [2], Semantic Databases [22], Cloud Databases [10] and NoSQL Databases [58]. The idea of using materialized views to satisfy the quality-of-service of databases does not date from today, but since forty years [35]. Their importance has been amplified since data warehouse physical design has become more sophisticated to cope with complex decision support queries [14,4]. Selecting a set of materialized views that would satisfy functional and non-functional requirements is complex [24]. This gave rise to the materialized view selection problem (VSP). Due to the importance and complexity of this selection, several research efforts from academia and industry have been conducted. Certainly, the results obtained

by these efforts have been implemented in commercial and open-source DBMSs such as Data Tuning Advisor for SQL Server [1], Design Advisor for DB2 [64], SQL Access Advisor for Oracle, and Parinda for PostgreSQL [39].

By examining the major solutions of VSP, we figure out that they usually consider static workload of queries . In other terms, the potential views are quantitatively evaluated and then greedily pre-materialized prior to executing the query workloads [49]. Formally, the VSP is defined in the literature as follows: given a workload of queries $Q = \{Q_1, Q_2, \ldots, Q_n\}$ and a set of resource constraints C (e.g., storage cost and maintenance cost). The VSP consists in selecting a set of materialized views $MV = \{V_1, V_2, \ldots, V_m\}$ that satisfies some of the non-functional requirements such as minimizing query performance, saving energy consumption, etc. and respects C.

This situation is inadequate with the nowadays requirements of analytical applications, where *high number*, *dynamic*, and *high interacted queries* are *simultaneously considered*.

Due to the importance of the above three properties of nowadays workloads containing a high number of dynamic, and highly interacted queries, their clarification is necessary. With regard to the high number of queries, let us consider the following real examples covering analytical and semantic databases: **(i)** the Periscope application manages twenty-something million queries per day[3]; **(ii)** the snowflake platform deals with more than 300 million queries per day from its customer base[4], and **(iii)** the obtained results of a recent paper published in VLDB'2020, dealing with the problem of selecting materialized views in Oracle DBMS, are obtained based on 650 queries running on a star schema [2], and **(iv)** the SPARQL query logs executed at scholarly data of DBpedia contains 43 284 queries [34].

The query operation sharing is a guiding characteristic and at the same time impacted by the two other properties. Sharing computation among multiple concurrent queries was first studied by Sellis in 1988 in the context of multi-query optimization (MQO). Recently, this principle has been reproduced in the context of Cloud Databases under the name "*Pay One, Get Hundreds for Free*" [41]. The identification of common subexpressions of queries is the key issue for the performance of multi-query processing, *even for a small set of queries*, which was the natural hypothesis of existing studies. Historically, the Problem of Multi-Query Optimization (PMQO) has been largely studied in the 80's [13], [56] in the context of relational databases [50]. This problem has been revisited in all database generations without any exception since it aims at optimizing the global performance of queries collectively instead of individually. The PMQO has been combined with several important database problems such as caching [46], materialized view selection [40], reusing [23], indexing [28], data partitioning [5], optimizing exploratory queries [32]. In the context of relational data warehouses, the PMQO has been amplified since OLAP workloads are a windfall of query sharing, where typical OLAP and reporting workloads are overlapping.

Regarding the dynamic aspect of OLAP queries, the diversity of data analysis and the changing business directives make the query workload more dynamic [53]. This dynamicity has contributed to increasing the research studies on self-tuning and autonomous databases [2,11,47]. Several reactive approaches such as *DynaMat* [33], *WATCHMAN*

[3] https://thenewstack.io/how-periscope-uses-kubernetes-to-power-data-science-services/
[4] https://diginomica.com/snowflake-ceo-insists-his-company-can-take-heat

[54], and Materialized Query Table (MQT) advisor [49] for dynamic materialized view selection have been proposed. The main characteristic of these approaches is that they react to transient usage, rather than purely relying on a historical workload [48]. These solutions have to solve multiple problems [37]: what views to materialize [49] for serving current and future queries, when to evict views (LRU cache) [49].

This motivates us to propose in this paper, a Proactive Re-selection of materialized views (called ProRes) that integrates concurrently our three properties (Figure 1). Since our queries arrive dynamically, based on a threshold, the first δ incoming queries are routed to the offline phase that selects the most beneficial common subexpressions for materialization purposes. The other coming queries are managed by the online phase that exploits the selected views of the offline one. *ProRes* integrates three fundamental aspects: **(i)** the usage of dynamic hypergraph structure that captures the interaction among queries, **(ii)** bounding intervals are used to update the current set of materialized views based on their benefits, and **(iii)** query scheduling if necessary. Contrary to most traditional studies which assume that the queries are already pre-ordered, our queries can be scheduled if their order reduces the query performance.

The paper is organized as follows: Section 2 overviews the most important studies related to our two studied problems: PMQO and VSP. Section 3 presents the basics and definitions related to hypergraphs and the processes for passing from queries to hypergraphs. Section 4 describes our ProPres approach, where all its components are detailed. Section 5 presents our intensive experiments and a real validation in commercial DBMS. Section 6 concludes the paper and discusses future work.

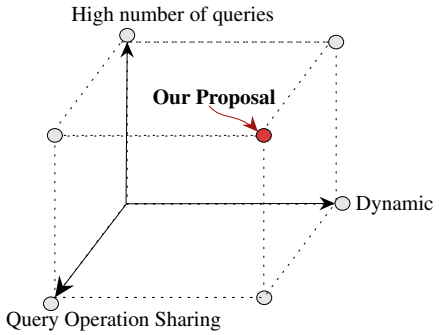


Fig. 1. Position of our Proposal according to the state-of-the-art

2. Related Work

This section discusses the most important studies dealing with PMQO and VSP either in isolation or jointly.

MQO is one of the main topics studied by the database community [52]. A specific chapter on MQO has been reserved in the encyclopedia of Database Systems edited by Ling Liu and Tamer Özsu [38]. Before connecting MQO to VSP, it would be wise to provide its separate generic formalization. For a given workload of queries to be optimized, where each query has a set of individual plans, the PMQO aims at finding the

best merging of query plans where the global query processing cost is minimized. The PMQO has proven as NP-hard in [59] where its search space formulation was given. An A* algorithm with bounding functions and intelligent state expansion, based on query order, to eliminate states of little promise rapidly has been proposed and guarantees an optimal solution, for a small set of queries [56]. Several variations of Sellis's algorithms have been proposed [59] to prune the search space of the PMQO. Genetic, simulated annealing and Depth-first Branch-and-Bound algorithms have also been experimentally analyzed to handle larger MQO problems that cannot be solved using A* in a reasonable time [16] [57]. Other algorithms are based on game theory [3]. [52] address the problem of extending top-down cost-based query optimizers to support multi-query optimization, and present greedy heuristics, as well as implementation optimizations. Their techniques were shown to be practical and to give good results. The adaptation of these findings has been reproduced in several generations of databases through their query languages: object-oriented databases (OQL) [63], semantic databases (SPARQL) [36], XML databases (XPATH) [20], distributed databases [32] [42], stream databases (CQL) [18], graph databases (Sparql 1.1) [19], data mining [44] and SQL-on-Hadoop systems [15].

MQO and physical design are the most active research topics in the field of databases and information systems. These two problems interact with the use of the query interaction which has a great impact on logical and physical optimizations. Despite their strong dependency, MQO and the instances of physical design have been tackled separately without really taking into account their interaction. The PMQO has usually been studied for a static set of queries, where several variations of the A* algorithm have been proposed for finding optimal solutions to the moderately sized (up to ten queries) [57]. Other studies advocated the fact that rather than trying to obtain an optimal solution, finding near-optimal solutions in less time is suitable. In the last years, there has been a growing interest in solving physical design problems especially in selecting materialized views, which are considered as one of the most interesting techniques to optimize OLAP queries. The VSP has been addressed in static and dynamic contexts without really considering their interaction with the MQO problem. To the best of our knowledge, The work proposed by [60] is the pioneer in the DW context that has highlighted this dependency. The main drawback of this work concerns the scalability of their algorithms in constructing the MVPP. To fill this gap, hypergraphs have been proposed in [8] to capture the query interaction among very large sets of static queries, then to study their contributions for selecting an appropriate set of materialized views. Promising results have been obtained showing the great benefit of hypergraphs to deal jointly with PMQO and the VSP [8] [51].

In physical design, Materialized views are one of the most important techniques to optimize analytical queries and are strongly dependent to query operation sharing. The materialized views selection problem is an NP-hard problem [24]. A large panoply of algorithms has been proposed to deal with this problem. We are not overviewing them, since several surveys exist. we suggest to readers the reference [51] which provides a nice classification of the existing algorithms. For complete classification of these algorithms, we recommend the readers to refer to the survey paper of [40] which divides algorithms into the following categories: deterministic algorithms, randomized algorithms, evolutionary algorithms, and hybrid algorithms. From the scope of our paper, these existing studies consider a small set of static queries, which contradicts the ad-hoc nature of analytical queries. Dynamat system [33] is one of the most popular systems that studied VSP in a

dynamic context. It monitors permanently the incoming queries and uses a pool to store the best set of materialized views based on a goodness metric and subject to space and update time constraints.

By analyzing the VSP studies, we realize that they are not connected to PMQO. The work proposed in [60] is the pioneer in the context of DW that showed the strong dependency among PMQO and VSP. It aims at constructing a unified query plan for a given set of queries. At first, they select the individual join plan for each query. Afterward, these plans are merged in a unified query plan called MVPP represented by a directed acyclic graph and dedicated to the process of selecting materialized views. The main limitation of this work is related to the scalability of their algorithms in constructing the MVPP. To counter this problem, hypergraphs have been proposed in [7] for coupling PMQO and VSP by considering the high number of queries. The authors have used the hypergraph structure for the identification of common subexpressions among a very large set of queries and to study their contributions for selecting an appropriate set of materialized views. In fact, Considering a huge number of queries produces a massive number of views to materialize which violates the storage space constraint. Indeed, it is impossible to materialize all candidate views under such a situation. To handle this problem, the authors [7] put a strategy that aims to maximize the benefit of using the materialized views before their dropping. To do so, they proposed a new query scheduling policy that allows finding the best query order that produces the highest benefit of the materialized views set.

An in-depth analysis of the major studies dealing with PMQO and VSP allows identifying several common points. The main shared point among PMQO and VSP is the use of graphs theory and their data structure either for pruning their large-scale search spaces or to identify the common sub-expressions among queries. Three main graph data structures have been proposed to prune the search space of VSP problem: AND/OR viewgraph [24] [43] , data cube lattice [25] [61] [29] and MVPP [60]. For PMQO, query graphs have been used to represent a set of queries [13] by merging the individual query trees in a single unified query plan (UQP). The UQP spans four levels of nodes: selection, join, projection and aggregation.

Another point shared by VSP and PMQO is the use of query scheduling policies to improve their algorithms. To illustrate this point, several good examples can be given. For instance, the work of [16], where the query scheduling has played a crucial role to improve the MQO algorithm. Also, several works have studied VSP under query scheduling constraints. In [49], a dynamic formalization of VSP is given by considering query scheduling policy based on a genetic algorithm. The two main limitations of Phan's algorithm [49] are (i) their dependence on DB2 advisor and (ii) their greedy genetic algorithm for reordering queries which are not suitable when scaling. To overcome these limitations, a scalable approach called *SLEMAS* is proposed in [7]. The scalability of this approach is ensured by the means of hypergraphs structure which is used to identify the query operation sharing among very large sets of queries. Afterward, the most shared operations are considered as candidates for materialization. The materialized views in SLEMAS are dynamically selected by considering scheduling constraints for a priori known workload of queries. The main drawback of this approach is their assumption of static sets of queries, which contradicts the dynamic nature of analytical queries.

Recently, PMQO and VSP are studied and revisited under a new angle by considering simultaneously the 3-characteristics of today's analytical queries [45]. To do so, dynamic

hypergraphs have been used to capture the query operation sharing, and dynamically selecting the appropriate set of materialized views without any consideration of the query scheduling constraint. Intensive experiments are conducted by this work [45] to compare the efficiency of their proposal against the major state-of-art. The obtained results showed the great impact of the query scheduling techniques in maximizing the benefit of materialized views and optimizing the performance of incoming workloads. Table 1 summarizes our discussion by showing the interest of the main studies dealing with PMQO and VSP in an isolated way or jointly.

Table 1. Classification of existing works on PMQO and VSP

Problems	**Work**	**Used Properties**	**Data Structure**	**Query Scheduling**	**Drawbacks**
MQO	[56]	Query Sharing	Query Graph	No	Scalability
	[16]	Query Sharing	Query Graph	Yes	Scalability
VSP	[33]	Dynamic	No	No	Scalability
	[49]	No one	No	Yes	Scalability and DB2 Dependence
MQO and VSP	[60]	Query Sharing	MVPP	No	Scalability
	[7]	High number of queries & Sharing	Hypergraph	Yes	Static
	[45]	three-properties	Hypergraph	No	Pre-ordered Queries

3. Background

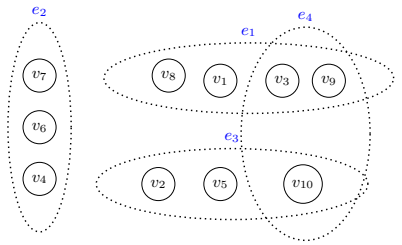
In this section, we present some fundamental notions and definitions related to hypergraphs and their ability to manage the three properties of analytical queries.

Hypergraphs are powerful tools for representing complex and non-pairwise relationships. They contributed in several domains in capturing the interaction between studied objects such as data mining, text/image retrieval, bio-informatics, social mining, and machine learning [27]. In the database fields, Hypergraphs have been used at logical and physical phases (e.g., the detection of functional dependencies [21], data partitioning for optimizing OLTP workloads [17] and materialized views selection for optimizing large-scale OLAP workloads [8]).

Definition 1. *A hypergraph $H = (V, E)$, is defined as a set of vertices V (nodes) and a set of hyper-edges E, where every hyper-edge connects a non-empty subset of nodes [9]. Note that when $|e_i| = 2$ $(\forall i = 1...m)$, the hypergraph is a standard graph.*

Definition 2. *The degree of a vertex $v_i \in V$, denoted by $d(v_i)$ represents the number of distinct hyper-edges in E that connect v_i.*

The incidence matrix of a hypergraph allows counting the connection between hyper-edges and vertices, where rows and columns represent respectively vertices and hyper-edges. The $(i, j)th$ value in the matrix, denoted by $IMij$, is equal to 1 if vertex v_i is



(a) Hypergraph H_1

$$\begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \\ v_{10} \end{array} \begin{array}{c} \begin{array}{cccc} e_1 & e_2 & e_3 & e_4 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix} \end{array}$$

(b) The Incidence Matrix of H_1

Fig. 2. Example of Hypergraph

connected in the hyperedge e_j, and 0 otherwise.

$$IMij = \begin{cases} 1, & \text{if jth hyperedge contains the ith vertex} \\ 0, & \text{otherwise.} \end{cases}$$

3.1. Representation of a Query by a Hypergraph

For giving a realistic hypothesis, we consider in our study the SPJ (Select-Project-Join) class of queries, which represent the most common queries studied in database theory. A particular focus on selections and joins known as costly operations.

Definition 3. *A query tree is a tree data structure representing a relational algebra expression. The tables of the query are represented as leaf nodes. The relational algebra operations are represented as the internal nodes. The root represents the query as a whole.*

In our study, the query plan of a given query is obtained by left-deep tree [26], where all selections are pushed down as far down through its query graph (tree).

In the following, we show how a query tree is transformed into a hypergraph.

Example 1. To illustrate how to represent an OLAP query by a hypergraph, let assume the following query Q defined on the star schema benchmark (SSB)[5] that contains a fact table *Lineorder* and four dimension tables *Customer*, *Supplier*, *Part*, and *Dates*.

```
select  d_year, s_nation, p_category,
sum(lo_revenue - lo_supplycost) as profit
from     DATES, CUSTOMER, SUPPLIER, PART, lineorder
where  lo_custkey = c_custkey (J2)
and lo_suppkey = s_suppkey    (J1)
and lo_PARTkey = p_PARTkey    (J4)
and lo_orderdate = d_datekey  (J3)
and c_region = 'EUROPE'
and s_region = 'EUROPE'
and d_year = 1993
and p_mfgr = 'MFGR#2'
group by  d_year, s_nation, p_category
order by  d_year, s_nation, p_category;
```

The query tree of the query Q is given in Fig.3, where four joins and selections are well represented. This tree can be easily transformed to a hypergraph with four vertices representing the four joins $\{J_1, J_2, J_3, J_4\}$ and one hyperedge e corresponding to our query Q. Note that each join node is defined by its join predicate and its associated selections.

[5] http://www.cs.umb.edu/~poneil/StarSchemaB.pdf

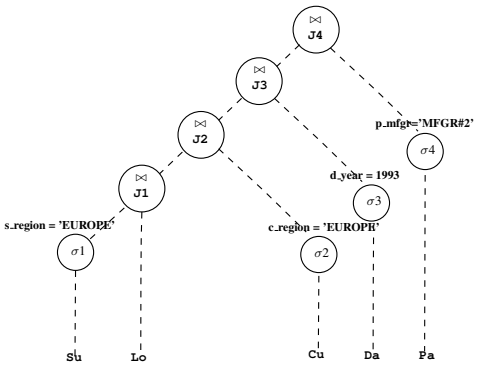
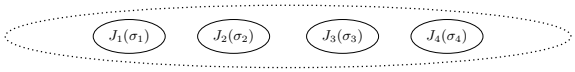


Fig. 3. The Tree of the Query Q



(a) Hypergraph of the Query Q

	e
J_1	1
J_2	1
J_3	1
J_4	1

(b) Its Incidence Matrix

Fig. 4. The Representation of the Query Q by a Hypergraph

From the query hypergraph, the algebraic tree can also be generated if the type of join processing tree (e.g., left deep, right deep, and bush) and the join order are a priori known. It should be noticed that the join order has a crucial role in optimizing star join queries involving dimension tables and a fact table. In our study, the fact table of an analytical query is always joined with dimension tables following their size (from small to large). Recent machine and deep learning-driven techniques for tackling the join order problem can be easily incorporated in our approach [62].

3.2. Hypergraph for Capturing the Interaction of Queries

In this section, we show how hypergraphs can easily capture the query sharing. For a given workload of queries W, we define a global hypergraph GH with a set of vertices GH_V and a set of hyperedges GH_E. Each vertex $v_i \in GH_V$ corresponds a join node J_i, whereas each hyperedge $e_j \in GE_E$ corresponds to a query $Q_j \in W$. The set of vertices of an hyperedge e_j represents the set of joins that participates in the processing of the query Q_j.

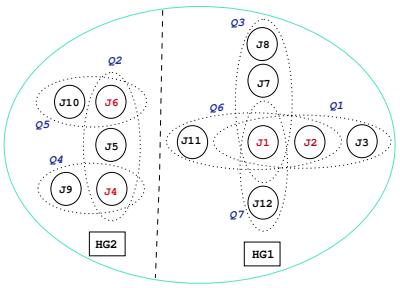
Definition 4. *A pivot node of a hypergraph is the first join shared by all queries.*

Example 2. To illustrate the construction of the global hypergraph for a given query workload, let us consider 7 OLAP queries ($\{Q_1, Q_2, Q_3, \ldots, Q_7\}$) defined in the appendix and generated randomly using the star schema benchmark query generator. Fig. 5 shows the obtained global hypergraph and its incidence matrix. The incidence matrix can easily give hints on the most shared joins by adding a column representing the usage frequency of

each join operation J_i ($Freq_i$):

$$\sum_{j=1}^{n} IM_{ij}. \tag{1}$$

Since the selection operations are performed before joins, our global hypergraph may contain several disjoint components. This is because selections reduce query sharing. In our example, the hypergraph contains two components including respectively $GH_1 : (Q_1, Q_3, Q_6, Q_7)$ and $GH_2 : (Q_2, Q_4, Q_5)$. We observe that all queries of the first component GH_1 share the same join operation identified by node J_1. As shown in the hypergraph and its incidence matrix, the four nodes identified by J_1, J_2, J_4 and J_6 are the most shared join operations and they will be considered as candidates for materialization.



(a) The Global Hypergraph

	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7	FRQ
j_1	1	0	1	0	0	1	1	4
j_2	1	0	0	0	0	1	0	2
j_3	1	0	0	0	0	0	0	1
j_4	0	1	0	1	0	0	0	2
j_5	0	1	0	0	0	0	0	1
j_6	0	1	0	0	1	0	0	2
j_7	0	0	1	0	0	0	0	1
j_8	0	0	1	0	0	0	0	1
j_9	0	0	0	1	0	0	0	1
j_{10}	0	0	0	0	1	0	0	1
j_{11}	0	0	0	0	0	1	0	1
j_{12}	0	0	0	0	0	0	1	1

(b) The Incidence Matrix

Fig. 5. The Global Hypergraph of the 7 Queries and its Incidence Matrix

3.3. Hypergraph for Managing High Number of Queries

The hypergraphs theory has a long history in solving many large-scale problems thanks to their ability in modeling any relationships, and their efficiency in dividing large search space of difficult problems into several sub search spaces through their fast partitioning tools. The hypergraphs have shown efficiency in managing several applications dealing with a huge amount of data and transactions. Several real examples can be given: As in VLSI design, in which hundreds of millions of gates are needed to design logical circuits through the hypergraph. Also, in social network applications, hypergraphs have been widely used to capture the interaction and behaviors of users [65].

Hypergraph partitioning is commonly used in dividing the search space of combinatorial large-scale problems into several sub-search spaces, which reducing prominently the complexity of the studied problems. This feature is ensured through the advanced tools of partitioning, which ensure scalability. Hypergraph partitioning problem consists of dividing the vertex set of the hypergraph H into a fixed number of k disjoint partitions of bounded size $\Pi = \{P_1, P_2, \ldots, P_k\}$, while minimizing a given objective function [55].

hMeTis [31], and PaToH [12] are two examples of hypergraph partitioning techniques initially developed in the VLSI domain.

Figure 6 shows an example of partitioning of the hypergraph given in Figure 2a into three partitions.

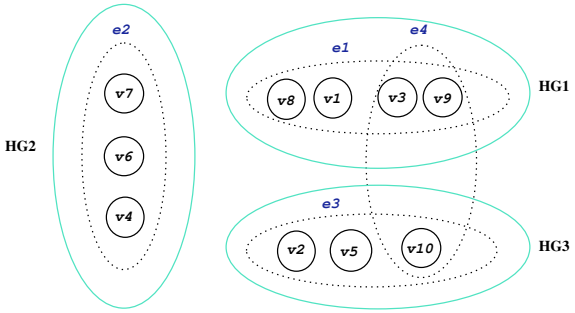


Fig. 6. An Example of Hypergraph Partitioning using hMetis

4. A Proactive Re-Selection of Materialized Views

We have all ingredients, to present our proactive re-selection of materialized views that considers our three properties. Before detailing our proposal, let us formalize our problem: given: (i) a set of queries known in advance, (ii) a set of ad-hoc queries that arrives dynamically, and (iii) a storage constraint. Our problem consists in selecting a set of materialized views speeding up the performance of all queries and satisfying the storage space constraint.

Our approach to deal with this problem is composed of two phases: offline phase that manages a set of the first δ queries Q_{first} based on the defined threshold δ and online phase that deals with the arrival of the ad-hoc queries. The global architecture of our proposal is described in Fig.7.

4.1. The Offline Phase

The Offline Phase is relies on the following main modules.

1. **Query parser**: allows parsing the set of queries Q_{first} in order to identify their logical operations (nodes) (Selection-Projection-Join).
2. **Hypergraph Construction:** once all queries of our first set of queries Q_{first} are parsed, we use the same rules described in Example 2 to construct our initial global hypergraph using two main primitives : add-node () , add-hyperedge().
3. **Hypergraph Partitioning:** Since our goal is to select joins that have a high sharing that are candidates for materialized. Therefore, our hypergraph has to be partitioned into groups of queries according to their interaction, where the query interaction is maximal inside each component and minimal among components. To ensure scalability, we adapt hMeTiS algorithm [30].
   Contrary to the original codes of hMeTiS, where the number of partitions is known in advance, in our case, the number of components to construct is unknown.
   To partition our hypergraph, we adapt an existing algorithm derived from graph theory to aggregate the join nodes into small connected components. The partition process is applied on initial hypergraph $GH(V, E)$ and the result of hypergraph partitioning is k sub-hypergraphs, where each one is an hypergraph: $GH_i(V_i, E_i)$, where $|E_i| \leq M$ $(1 \leq i \leq k)$. Our partitioning algorithm follows the same heuristic detailed in [6] that includes the following steps :

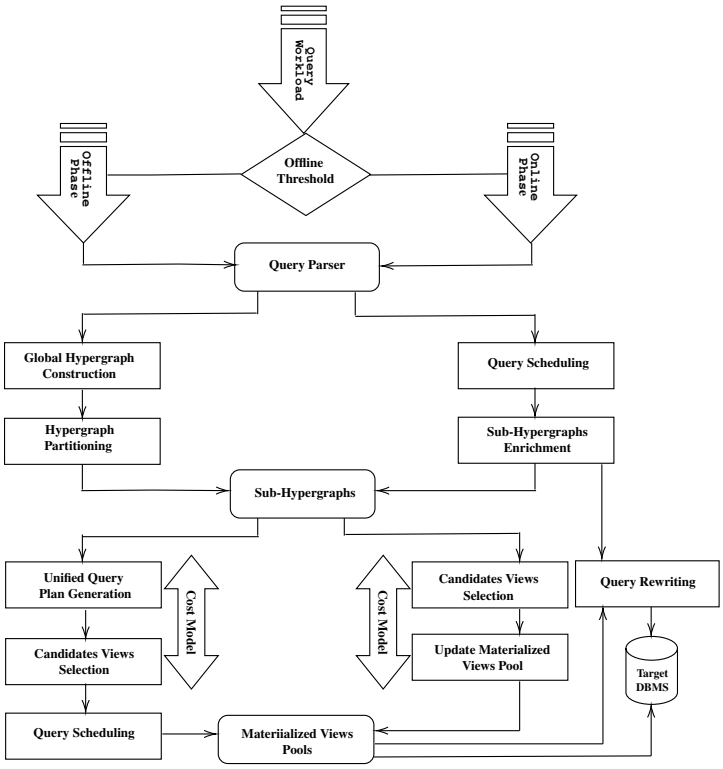


Fig. 7. The Global Architecture of our Approach

(a) Firstly, we adapt *the code of the multilevel hypergraph partitioning* Hmetis to split our hypergraph into k partitions such that the number of hyperedges cut is minimal. In our context, the exact number of partitions to construct is unknown. In the same time, we want to get all possible disjoint partitions (connected components). To do so, we adapt the original algorithm to our problem by bi-partitioning until no partition can be repartitioned without cutting hyperedges. More precisely, the algorithm behaves as follows: (i) the set of vertices will be divided if and only if the number of hyperedges cutting is null. (ii) Each bisection result of the hypergraph partitioning will be divided in the same way until no more divisible hypergraph is found.

(b) Secondly, we use Hmetis to partition each sub-hypergraph $GH_i(V_i, E_i)$, such as $|E_i| \leq M$. The sub-hypergraph $GH_i(V_i, E_i)$ is then partitioned into k' partitions such as $k' = (|E_i|/M) + 1$.

Table 2 summarizes the mapping between the *graph* vision and the *query* vision.

4. **Unified query plan Generation:** This step aims at generating the unified query plan (UQP) for each connected component by transforming each sub-hypergraph into an oriented graph. This generation is driven by cost models that allow ordering nodes. This step is necessary to order the join nodes in each component. Adding an arc to the oriented graph corresponds to putting an order between two join nodes. The transformation process has three main stages:

Table 2. Analogy Graph – Query.

Vision hypergraph	Vision of query
Set of vetices	Set of join nodes
Hyperedge	Query
sub-hypergraph	connect component
Oriented graph	Processing Plans

(a) choose the pivot node, the pivot node corresponds to the node which has the best possible benefit from reusing the intermediate results. The benefit of each node n_i is calculated using equation 2:

$$benefit(n_i) = (nbr\_use - 1) \times process\_cost(n_i) - constr(n_i) \quad (2)$$

where nbr_use, $process\_cost(n_i)$, and $constr(n_i)$ represent respectively the number of queries that use the join node ni, the processing cost of n_i, and the construction cost of n_i. In our study, we assume that the hash join is used to process join operations. The cost of a join involving two tables T_i and T_j is given by the following formula: $3 \times (|T_1| + |T_2|)$, where $|T_1|$ represents the number of pages of table T_1. The cost of construction of a node n_i is defined as a summation of the cost of its generation and storage.

(b) Transform the pivot node from the hypergraph to the oriented graph.

(c) Remove the pivot node from the hypergraph. We mention that we were inspired by the work proposed by [8] to generate the UQP which ensured the scalability of our approach. Figure 8 shows an example for the transformation step of the hypergraph to an oriented graph. In the end, the join nodes having a positive benefit are selected as candidates for materialization. In this example, the two nodes $J1$ and $J2$ are selected as candidates' views.

5. **Query Scheduling:** To increase the benefit and reusing of materialized views before their dropping, we propose to reschedule the queries of the set Q_{first}. The scheduler has the following formalization:
For a given hypergraph component, this module takes as an input the set of queries of this component and their join nodes already selected as candidates for materialization. Our scheduler module aims at providing scheduled queries in a new order maximizing the net benefit of using materialized views and reducing the overall processing cost of queries. In fact, we are inspired by the work proposed in [7] showing the efficiency of their scheduling algorithms in maximizing the benefit of materialized views and improving the query processing cost. Contrary to this algorithm in which the materialized views are all dropped after their usage by the appropriate queries. In our proposal, we keep the Pivot node from each component to serve the future incoming queries. The pivot nodes represent the views having the maximal benefit for each component. To clarify our scheduling process, let us consider the hypergraph component illustrated in the figure 9 in which 4 queries are involved. In this example, two join nodes are selected as candidates for materialization: $\{J1, J_2\}$ which are written in red. The first step of our process is to order the join nodes according to their benefits. Secondly, each query is assigned to a weight calculated by summing the benefit of its used queen nodes. Finally, we order queries according to their weights.

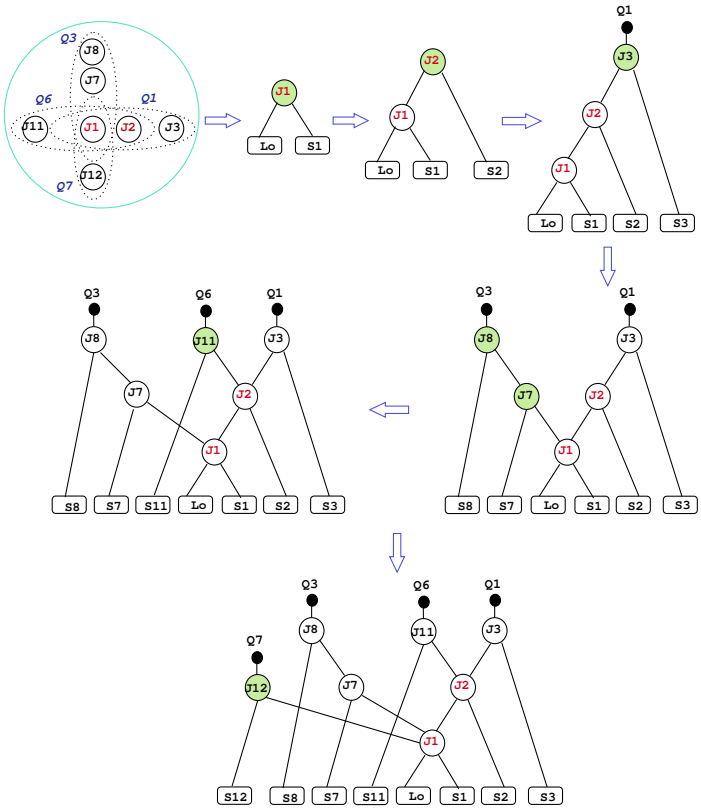


Fig. 8. From Hypergraph to Unified Query Plan

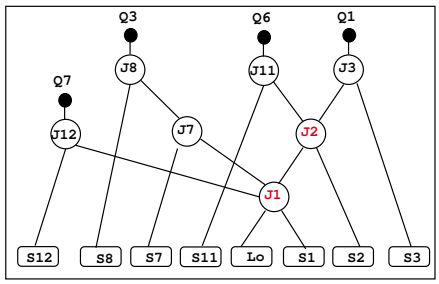
4.2. The Online Phase

In this section, we discuss our process for optimizing dynamic coming queries through our hypergraph structure. For managing the dynamic arrival of ad-hoc queries, a set of primitives are proposed for incrementally augmenting the global hypergraph constructed in the offline phase: add-node(), add-hyperedge(),remove-node(), remove-hyperedge(), etc. During the arrival of these queries, a set of materialized views is dynamically selected based on the identified shared joins through our hypergraph and a benefit function taking into account storage and maintenance constraints. At each instant t of the arrival of queries, the content and the size of the global hypergraph and the pool of materialized views will change dynamically. Two main modules characterized the online phase.

Enrichment of the Hypergraph To ensure an efficient enrichment of the global hypergraph, we put forward a strategy that consists in placing the incoming queries in the appropriate component and materializing the most shared joins identified dynamically by our hypergraph for re-using them by future queries. Our strategy for placing these queries is based on two criteria defined between the incoming query Q^t and the existing sub-hypergraphs. This is done by respecting the following principles:

Node	Benefit
J1	500
J2	100

Queries	Used Nodes	Weight
Q1	J1,J2	600
Q3	J1	100
Q6	J1,J2	600
Q7	J1	100



New Query Order	Q1	Q6	Q3	Q7

Fig. 9. Query Scheduling process

Priority 1: In order to increase the reuse of materialized views already selected and stored in the pool of each sub-hypergraph, we have defined a metric for the coming query Q^t called Nbr_Shared_Views and defined as follows .
$Nbr\_Shared\_Views(Q^t, GH_i^t) := |Joins(Q^t) \bigcap Pool^{GH_i^t}|$

The above metric represents the cardinal of the intersection among the set of join nodes of incoming query Q^t and the set of nodes already materialized for the ith sub-hypergraph GH_i^t. After calculating this metric among Q^t and each sub-hypergraph. The query Q^t has to be placed in the component that maximizes this metric.

Priority 2: If the query Q^t does not share any materialized views with the existing sub-hypergraphs, Q^t is placed in the component that shares with it the maximum of join nodes. To do that, we have defined a new metric called $Shared\ Joins\ Weight\ (Q^t, GH_i^t)$ (implemented in the algorithm2). This metric is calculated among the query Q^t and each sub-hypergraph. The metric $Shared\ Joins\ Weight$ is related to the degree of each join node belongs to Q^t in the incidence matrix of a given sub-hypergraph. The node's degree is returned from the last column of each incidence matrix that we name it FRQ column. To define the the metric $Shared\ Joins\ Weight$, we have to firstly define the degree function that allows calculating the join nodes degrees of Q^t in the incidence matrix of the ith component using algorithm 1.

Algorithm 1: Degree

Inputs a join node: V_i^t, the incidence matrix: $M^{GH_i^t}$;
Outputs the calculated degree: $Degree$;
if $V_i^t \notin nodes(GH_i^t)$ **then**
    $Degree$=0;
    /* if the vertex V_i^t does not belong to the set of vertices of the sub-hypergraph (GH_i^t), then its degree is zero */
else
    $Degree = \sum_{j=1}^{Number\_Columns-1} M^{GH_i^t}(i,j)$;
    /* We assume that the join node Vi is positioned in the ith Row */
end

Using the above degree algorithm, we have defined the algorithm $Shared\_Joins\_Weight\ (Q^t, GH_i^t)$ that calculates the weight of the incoming query Q^t in the incidence matrix of the ith hypergraph component GH_i^t.

Algorithm 2: Shared Joins Weight

Inputs The incoming query: Q^t, the ith sub-hypergraph: GH_i^t;
Outputs the calculated weight: $Shared\_Joins\_Weight$;
$Shared\_Joins\_Weight \leftarrow 0$;
foreach $node \in join\_nodes(Q^t)$ **do**
    $Degree \leftarrow Degree(node, M^{GH_i^t})$;
    if $Degree \neq 0$ **then**
        $Shared\_Joins\_Weight \leftarrow Shared\_Joins\_Weight + 1$;
    end
end

The dynamic changes in the hypergraph components impose us to dynamically updating their incidence matrix. To do so, we have defined the algorithm Update Incidence Matrix $(Q^t, M^{GH_i^t})$ that allows adding columns, rows and updating the FRQ column which represents the degree of nodes in the incidence matrix. We perform this task when a new query is placed in the ith hypergraph component. If the query Q^t does not share any join with the existing components, a new hypergraph component is constructed and associated with this query.

Algorithm 3: Update Incidence Matrix

Inputs The incoming query : Q^t , the incidence matrix: $M^{GH_i^t}$;
Outputs The updated incidence matrix: $M^{GH_i^t}$;
add the column Q^t to the matrix $M^{GH_i^t}$;
foreach $node \notin nodes(Q^t)$ **do**
    if $node \notin nodes(GH_i^t)$ **then**
        add new row to the matrix $M^{GH_i^t}$;
        $Degree(node, M^{GH_i^t}) = 1$;
    else
        $Degree(node, M^{GH_i^t}) = Degree(node, M^{GH_i^t}) + 1$;
    end
end

Dynamic Re-Selection of Materialized Views In fact, the dynamic and continuous arrival of queries produces a massive number of views to materialize which violates the storage space constraint. Indeed, it is impossible to materialize all candidate views under such a situation. To cope with this problem, we put forward a strategy that consists in materializing the most shared joins that can be used by future incoming queries. To do that, we repeat the following process at each arrival of a new query to a component until the saturation of the fixed storage space: At the arrival of a new query Q^t to the ith hypergraph component GH_i^t, we calculate the benefit of each joins belonging to the query

Q^t using our benefits function 2. Afterward, we check if there are joins having a positive benefit. In this case, we call the algorithm $Update\ Materialized\ Views$ to update the pool of the ith component GH_i^t by materializing joins which have a positive benefit. If the storage space is saturated, we drop the materialized views having the least benefit in the pool of this component and we replace them by materializing beneficial joins of the current query.

Algorithm 4: Update Materialized Views

Inputs the query: Q^t; the hypergraph component : Fi;
the views pool associated to the Component Fi: $Pool^{F_i}$; disk Space: DS;
Output the updated pool of the component Fi: $Pool^{F_i}$;
$joins \leftarrow Join(Q^t)$;
Get_Benefit (Joins);
$L \leftarrow Return\_Joins\_with\_Positive\_Benefit(Joins)$;
Descending Order(L) ;
foreach $node \in L$ **do**
  if $node \notin Pool^{F_i}$ $And\ size(Pool^{F_i}) + size(node) < DS$ **then**
    Materializing (node);
    $Pool^{F_i}.add(node)$;
    $size(Pool^{F_i}) \leftarrow size(Pool^{F_i}) + size(node)$;
  else
    if $node \notin Pool^{F_i}$ $And\ size(Pool^{F_i}) + size(node) > DS$ **then**
      Ascending Order($Pool^{F_i}$) ;
      $idx \leftarrow 0$;
      repeat
        if $benefit(node) > benefit(Pool[idx])$ **then**
          Dropping (Pool [idx]) ;
          $size(Pool^{F_i}) \leftarrow size(Pool^{F_i}) - size(node)$;
          $idx \leftarrow idx + 1$;
        end
      until $Pool^{compi}) + size(node) < DS$ *OR* $benefit(node) < benefit(Pool[idx])$;
      if $size(Pool^{compi}) + size(node) < DS$ **then**
        Materializing (node);
        $Pool^{F_i}.add(node)$;
        $size(Pool^{F_i}) \leftarrow size(Pool^{F_i}) + size(node)$;
      end
    end
  end
end

Algorithm 5 describes our dynamic process for constructing the hypergraphs and selecting materialized views in the online phase. To illustrate our dynamic strategy for managing ad-hoc and dynamic queries, an example is given in the figure 10 by considering the same query workload and hypergraph considered in the previous examples 2.

Algorithm 5: Incremental Construction of the hypergraph And Dynamic Materialization

Inputs: the incoming query Q^t at the instant t; Storage space Sp;
Outputs: a list of components (F) of GH; pool of views for each component;
loop;
$t := 1$;
Query_Parser (Q^t);
if $|F^t| = 0$ **then**
    $F^t := Construct\_new\_component()$;
    $add\_edge(F^t, Q^t)$;
    $M^t := Calculate\_incidence\_matrix(GH^t)$;
else
    foreach $F_i^t \in GH^t$ **do**
        $Nbr\_shared\_views^t := |Joins(Q^t) \bigcap Pool^{F_i^t}|$;
        $List_1^t.add(Nbr\_shared\_views^t)$;
        $Shared\_Joins\_Weight^t := Shared\ Joins\ Weight(Q^t, F_i^t)$;
        $List_2^t.add(Shared\_Joins\_Weight^t)$;
    end
    $Maximum_1^t := Maximum(List_1^t)$;
    if $Maximum_1^t = 0$ **then**
        $Maximum_2^t := Maximum(List_2^t)$;
        if $Maximum_2^t = 0$ **then**
            $Construct\_new\_component(F_i^t)$;
            $add\_hyperedge(F_i^t, Q^t)$;
            $F.add(F_i^t)$;
            $M^t := Update\_incidence\_matrix$;
        else
            $pos^t := Return\_pos(Maximum_2^t, List_2^t)$;
            $add\_hyperedge(F.get(pos^t), Q^t)$;
            $UpdateIncidenceMatrix(Q^t, M^t)$;
            $UpdateMaterializedViews(Q^t, F.get(Pos^t), Pool^{F.get(Pos^t)}, Sp)$;
        end
    else
        $Pos^t := Return\_posn(Maximum_1^t, List_1^t)$;
        $add\_hyperedge(F.get(pos^t), Q^t)$;
        $Rewrite(Q^t, Pool^{F.get(Pos^t)})$;
        $UpdateIncidenceMatrix(Q^t, M^t)$;
        $Update\ Materialized\ Views(Q^t, F.get(Pos^t), Pool^{F.get(Pos^t)}, Sp)$;
    end
end
$t := t + 1$;
END LOOP

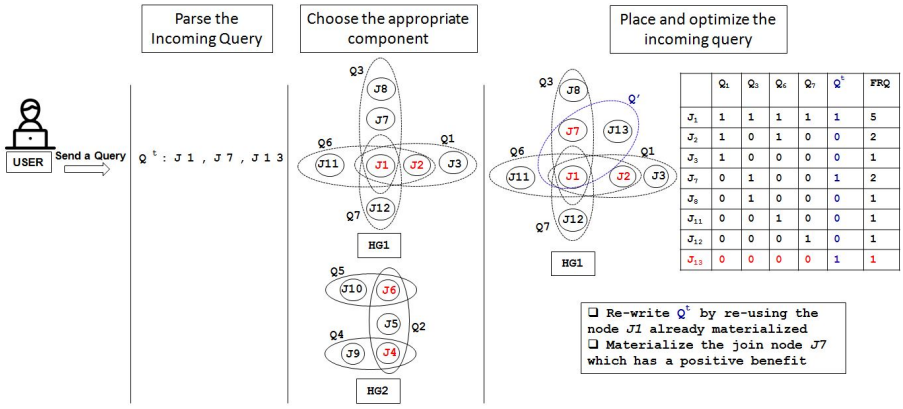


Fig. 10. The optimization process of an ad-hoc query

5. Experimental Study

In this section, we firstly show the connection of our approach to a commercial DBMS. Afterward, we conduct an efficiency study to validate and compare our proposal with major state-of-art studies.

5.1. ProRes Connection to Oracle DBMS

Based on our findings, we have developed a tool, called ProRes inspired by the well-known commercial advisors allows assisting DBA in their tasks when selecting materialized views with a strong advantage in managing the three-properties of analytical queries. ProRes is developed using Java and integrates all phases and modules of our approach. Our wizard is illustrated in fig 11.

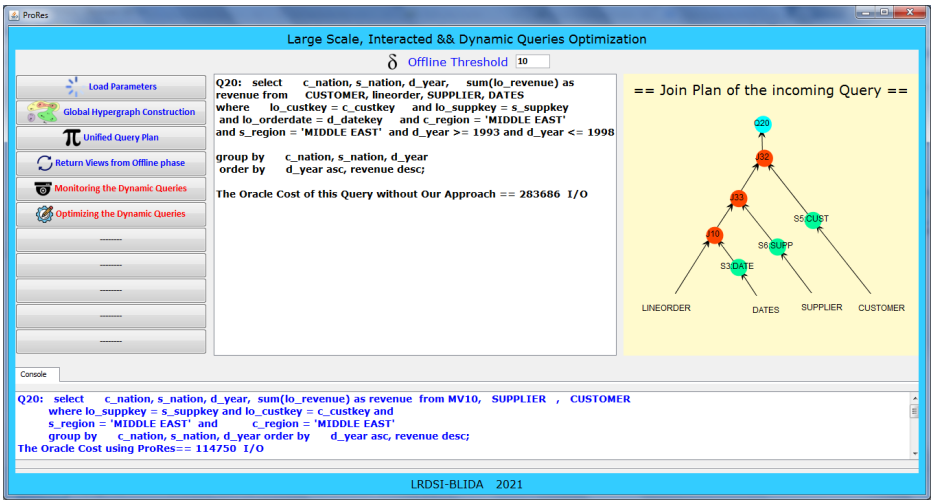


Fig. 11. An Example of Functioning of ProRes.

5.2. Efficiency Study

In this section, we present an experimental validation of our approach using the following environment: a server with E5-2690V2, 3 GHz processor, 24 GB of main memory, and 1 TB of the hard disk. We generate a DW with 30 GB deployed in Oracle 12c DBMS and queries using SSB generator modules.

Our approach has been compared against two approaches: **(1)** the approach proposed in [60] (that we name it *YANG*) which is considered the pioneer study in data warehouses that highlights the strong dependency among PMQO and VSP. For *YANG* we have developed both of their algorithms :(a) their naive algorithm called A feasible solution that generates all possible MVPP and choose the plan with minimum cost (b) their algorithm based on 0-1 integer programming which is faster than the first. **(2)** the algorithm proposed in [49] which is considered as one of the most important works that highlighted the crucial role that query scheduling plays in dealing efficiently with dynamic materialized views selection. We reference this work in our experiments by *PHAN*. For *PHAN*, we have developed the following algorithms : (a) their genetic algorithm which aims to find the optimal order of queries by using natural selection taken from Darwin's theory with 1000 generations. (b) an algorithm for selecting the nodes having the greater benefit as candidates for materialization (in [49], these nodes are selected using DB2 advisor) (c) their algorithm for pruning the set of candidates nodes based on their benefit (d) their algorithm for evaluating the net benefit of the pruned set of candidates views in optimizing a given query workload. (e) and finally, an algorithm for managing the cache following LRU rules.

Number of selected materialized views and their benefit Optimizing a big workload of queries by selecting a small set of materialized views is one of the crucial quality metrics recently highlighted by a leading DBMS editor [2]. Therefore , we attempt in this experiment to evaluate the three algorithms in terms of the number of selected materialized views, the number of optimized queries, and the number of dropped views. To do so, two experiments were conducted by considering two different workloads with 100 and 1000 queries. The obtained results are summarized in Fig.12. The obtained results show that our approach outperforms the other approaches in terms of the number of optimized queries.

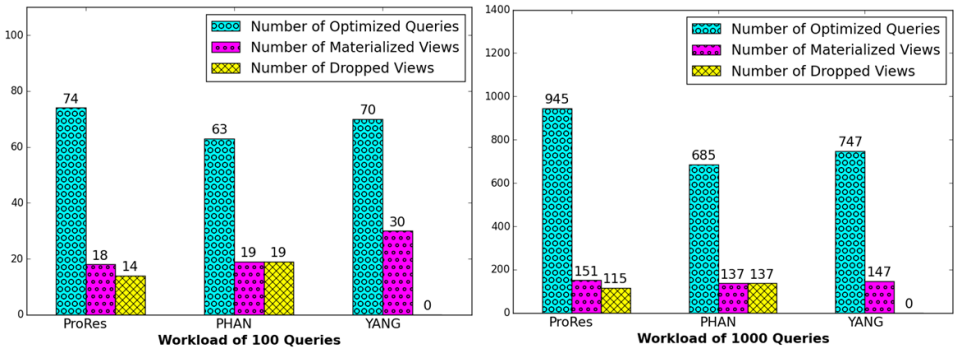


Fig. 12. A comparison among Our Approach, Phan and Yang algorithms

Impact of selected views on query processing and materialization costs The goal of this experiment is to study the contributions of the selected materialized views by our approach and the other algorithms on the overall query processing and their materialization costs. To do so, we have evaluated theoretically the different costs using our mathematical cost model. The obtained results are reported in Fig. 13. The selected views by our approach are more beneficial than those generated by the other algorithms. This is due to our materialization strategy that selects the most beneficial candidates and to our scheduling policy that allows augmenting the benefit of the selected materialized views.

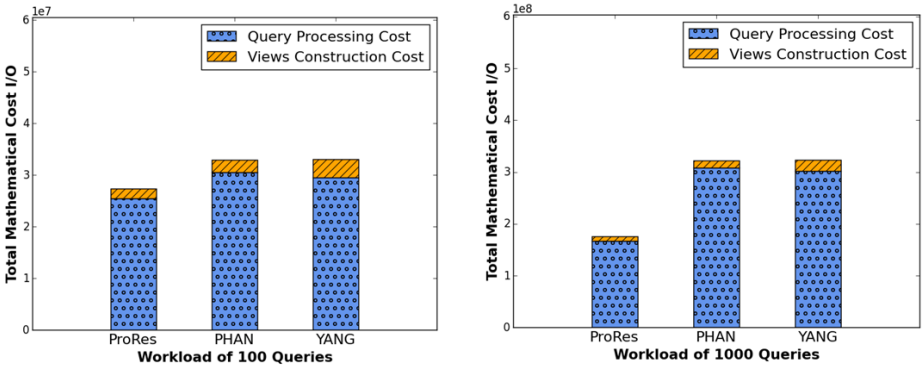


Fig. 13. Comparison among the three approaches in terms of processing/maintenance costs

Oracle Validation: Comparison of the three approaches The goal of this experiment, is to validate the results obtained theoretically in the previous experiments. To do this, we consider a workload of 100 queries running on a DW with 30 GB deployed in Oracle 12c DBMS. The storage space for materialized views is set to 60 GB. The obtained results are reported in Fig. 14. We observe that our algorithm outperforms the other algorithms. The obtained results coincide with those obtained theoretically, which confirms the efficiency and the superiority of our approach .

Impact of query scheduling and dynamic materialization on optimizing queries For testing the efficiency of our approach two scenarios are considered:

(a) Static Materialization: In this experiment, we consider a naive scenario in which the nodes selected by each algorithm are materialized till the saturation of the storage space. To perform this experiment, we have considered a data warehouse with 30 Gb and a workload of 100 queries using SSB Benchmark. As shown in Fig.15a, there is not a big difference between our approach and Yang's algorithm, which demonstrate that our approach does not avoid the selection of the best-materialized views.

(b) Dynamic Materialization with Query Scheduling: To perform this experiment, we have used the same above data by considering SSB data set with 30 Gb and a query workload of 100 queries generated randomly using the SSB generator. As shown in Fig.15b, our approach outperforms largely Phan's algorithm. This is due to the minimal

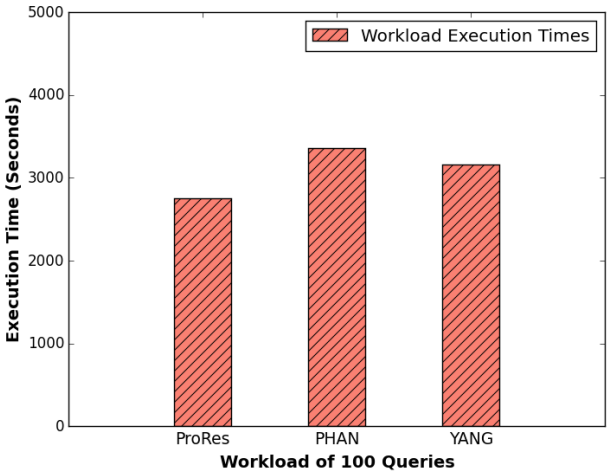
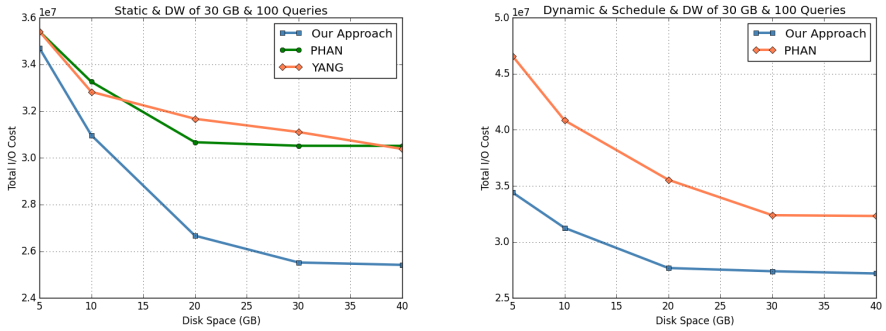


Fig. 14. Workload execution times.

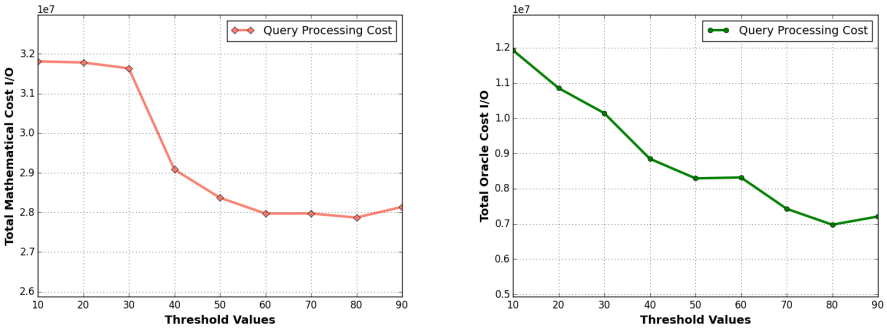
number of dropping in our approach than that of Phan. In addition, the materialized views selected by our approach are used maximally to optimize the appropriate queries before their dropping.



(a) Static scenarios **(b)** Dynamic scenarios

Fig. 15. Performance of our approach in static and dynamic scenarios

Impact of the threshold δ on the Processing Cost of Queries In this experiment, we first evaluate theoretically (using our cost model) then in Oracle 12c the overall processing cost of a given workload of 100 queries by varying the value of δ threshold. The obtained results are depicted in 16. We observe the obvious effect of the threshold δ on the query processing cost, where there is an inverse relationship between the threshold values δ and the processing cost of queries. At each increase in the threshold value, the processing cost of queries decreased until the threshold value reaches 70 which represents the stability point of the processing cost optimization.



(a) From theoretical Perspective **(b)** Oracle Validation

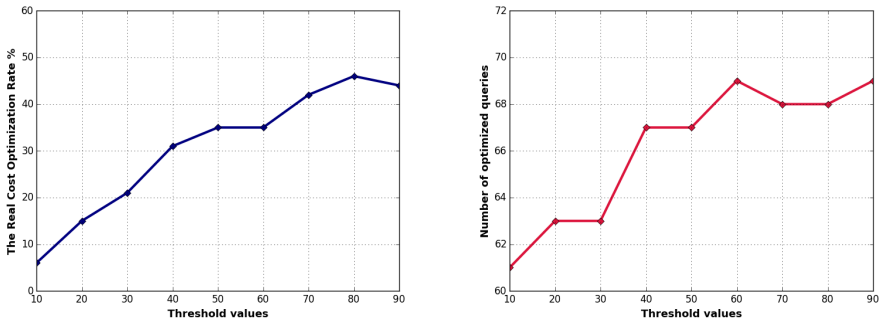
Fig. 16. The Effect of the Threshold on the Processing Cost

The processing cost reduction rate according to the threshold values In this experiment, we follow the same above scenario by considering the same set of 100 queries and varying the δ threshold values. The goal of this experiment is to evaluate the impact of the threshold values and the quality of the materialized views selected by our proactive strategy on the overall processing cost of queries. To do this, we firstly estimate the overall real processing cost of the query workload without using our approach (costwithout). Afterward, we estimate the processing cost of this workload using our approach by varying the offline threshold values. Finally, we compute the cost reduction rate as : $1 - \frac{query\ cost\ with\ views}{query\ cost\ without\ views}$.
Fig.17a shows the obtained results implemented in Oracle 12c DBMS. The obtained results confirm the previous results and prove that our approach becomes more interesting when the offline threshold increases until the stability point value $\delta \geq 70$, where the cost reduction rate is among 42% and 47%. This is due to the expansion of our pool by the most beneficial materialized views.

The Number of optimized queries according to the offline threshold values: In the last experiment, we attempt to evaluate our proactive strategy in terms of the number of optimized queries according the threshold values δ . To do this, we consider the same above scenarios and the same workload of queries. The obtained results are described in Fig17b. They showed the proportional Relationship among the number of optimized queries and the offline threshold values until the value $\delta \geq 60$, where there is a stability in the number of optimized queries, where 69 queries have been optimized from the 100 received queries.

6. Conclusion

In this paper, we discuss the opportunity offered by the best aspects brought by the era of Big Data to augment the data warehouses technology. Note that several efforts have been recently deployed to augment this technology. But, they did not give great attention to the high number of queries as the other aspects despite its strong connection to the different phases of the data warehouse life cycle. Today, analytical queries are known by three



(a) The Oracle cost reduction rate **(b)** The Number of optimized queries

Fig. 17. The cost reduction rate and the Number of optimized queries according to the offline threshold

main properties: (1) very numerous, (2) dynamic, and (3) share similar operations. These characteristics may touch several well-studied and known problems such as multi-query optimization (PMQO), physical design. The classical solutions cannot be used directly to handle these three properties during the physical and logical optimization of queries. We guess that dealing with these three properties requires the usage of a flexible data structure. Therefore, hypergraphs have been proposed for coupling PMQO and VSP under a new angle by considering our three properties. In this work, we were inspired by the aspects of new business intelligence (BI next-generation), where there are two types of queries: a priori known queries and ad-hoc queries. To deal with both types of queries, we have proposed a proactive approach composed of two phases: an offline phase for optimizing queries known in advance and an online phase dedicated to optimizing the ad-hoc queries. The offline phase and the online phase share the use of the same hypergraph structure for capturing the query interaction and selecting the candidate views. The main particularity of our approach is that it covers the two main categories of the dynamic selection of views. Our proposal implemented by a simulator called ProRes was validated theoretically using mathematical cost models and its results were directly implemented on the Oracle DBMS. The obtained results are encouraging and show the efficiency and effectiveness of our approach.

Our work opens several challenges: (i) currently, we are integrating our proposal into PostgreSQL DBMS, (ii) considering other optimization techniques such as horizontal data partitioning and indexes, (iii) integrating the machine learning techniques for predicting the appropriate optimization structure for the ad-hoc queries. (iv) reproduce our proposal for using hypergraphs to deal with SPARQL queries in semantic databases.

References

1. Agrawal, S., Chaudhuri, S., Kollár, L., Marathe, A.P., Narasayya, V.R., Syamala, M.: Database tuning advisor for microsoft SQL server 2005. In: VLDB. pp. 1110–1121 (2004)
2. Ahmed, R., Bello, R.G., Witkowski, A., Kumar, P.: Automated generation of materialized views in oracle. Proc. VLDB Endow. 13(12), 3046–3058 (2020)

3. Azgomi, H., Sohrabi, M.K.: A game theory based framework for materialized view selection in data warehouses. Engineering Applications of Artificial Intelligence pp. 125–137 (2018)
4. Bellatreche, L., Karlapalem, K., Schneider, M.: On efficient storage space distribution among materialized views and indices in data warehousing environments. In: ACM CIKM. pp. 397–404 (2000)
5. Bellatreche, L., Kerkad, A.: Query interaction based approach for horizontal data partitioning. IJDWM 11(2), 44–61 (2015)
6. Boukorca, A.: Hypergraphs in the Service of Very Large Scale Query Optimization. Application. Phd thesis, ISAE-ENSMA, Poitiers France (2016)
7. Boukorca, A., Bellatreche, L., Cuzzocrea, A.: SLEMAS: an approach for selecting materialized views under query scheduling constraints. In: International Conference on Management of Data (COMAD) . pp. 66–73 (2014)
8. Boukorca, A., Bellatreche, L., Senouci, S.B., Faget, Z.: Coupling materialized view selection to multi query optimization: Hyper graph approach. IJDWM 11(2), 62–84 (2015)
9. Bretto, A.: Hypergraph Theory: An Introduction. Springer (2013)
10. Bruno, N., Jain, S., Zhou, J.: Continuous cloud-scale query optimization and processing. Proc. VLDB Endow. 6(11), 961–972 (2013)
11. de Carvalho Costa, R.L., Moreira, J., Pintor, P., dos Santos, V., Lifschitz, S.: Data-driven performance tuning for big data analytics platforms. Big Data Research pp. 100–206 (2021)
12. Çatalyürek, Ü.V., Aykanat, C.: Patoh (partitioning tool for hypergraphs). In: Padua, D.A. (ed.) Encyclopedia of Parallel Computing, pp. 1479–1487. Springer (2011), https://doi.org/10.1007/978-0-387-09766-4_93
13. Chakravarthy, U.S., Minker, J.: Multiple query processing in deductive databases using query graphs. In: VLDB. pp. 384–391 (1986)
14. Chaudhuri, S., Narasayya, V.R.: Self-tuning database systems: A decade of progress. In: VLDB. pp. 3–14 (2007)
15. Chen, T., Narita, K.: Multiple query optimization in sql-on-hadoop systems. US Patent 10,572,478 (2020)
16. Cosar, A., Lim, E.P., Srivastava, J.: Multiple query optimization with depth-first branch-and-bound and dynamic query ordering. In: International Conference on Information and Knowledge Management(ACM-CIKM). pp. 433–438 (1993)
17. Curino, C., Zhang, Y., Jones, E.P.C., Madden, S.: Schism: a workload-driven approach to database replication and partitioning. PVLDB 3(1), 48–57 (2010)
18. Dobra, A., Garofalakis, M.N., Gehrke, J., Rastogi, R.: Sketch-based multi-query processing over data streams. Data Stream Management pp. 241–261 (2016)
19. Dobra, A., Garofalakis, M.N., Gehrke, J., Rastogi, R.: Multiple-query optimization of regular path queries. In: International Conference on Data Engineering (ICDE). pp. 1426–1430 (2017)
20. Fan, W., Yu, J.X., Li, J., Ding, B., Qin, L.: Query translation from xpath to SQL in the presence of recursive dtds. VLDB Journal 18(4), 857–883 (2009)
21. Fuentes, J., Sáez, P., Gutierrez, G., Scherson, I.D.: A method to find functional dependencies through refutations and duality of hypergraphs. Computer Journal 58(5), 1186–1198 (2015)
22. Goasdoué, F., Karanasos, K., Leblay, J., Manolescu, I.: View selection in semantic web databases. Proc. VLDB Endow. 5(2), 97–108 (2011)
23. Gupta, A., Sudarshan, S., Viswanathan, S.: Query scheduling in multi query optimization. In: The International Database Engineering And Applications Symposium (IDEAS). pp. 11–19 (2001)
24. Gupta, H., Mumick, I.S.: Selection of views to materialize under a maintenance cost constraint. In: The International Conference on Database Theory (ICDT). pp. 453–470 (1999)
25. Harinarayan, V., Rajaraman, A., Ullman, J.D.: Implementing data cubes efficiently. In: The ACM Special Interest Group on Management of Data (ACM-SIGMOD). pp. 205–216 (1996)

26. Ioannidis, Y.E., Kang, Y.C.: Left-deep vs. bushy trees: An analysis of strategy spaces and its implications for query optimization. In: The ACM Special Interest Group on Management of Data (ACM-SIGMOD). pp. 168–177 (1991)
27. Jiang, W., Qi, J., Yu, J.X., Huang, J., Zhang, R.: Hyperx: A scalable hypergraph framework. IEEE Trans. Knowl. Data Eng. 31(5), 909–922 (2019)
28. Jin, C., Carbonell, J.G.: Predicate indexing for incremental multi-query optimization. In: The International Symposium on Methodologies for Intelligent Systems (ISMIS) . pp. 339–350 (2008)
29. Kalnis, P., Mamoulis, N., Papadias, D.: View selection using randomized search. Data and Knowledge Engineering 42(1), 89–111 (2002)
30. Karypis, G., Aggarwal, R., Kumar, V., Shekhar, S.: Multilevel hypergraph partitioning: Application in vlsi domain. In: The Design Automation Conference (DAC). pp. 526–529 (1997)
31. Karypis, G., Aggarwal, R., Kumar, V., Shekhar, S.: Multilevel hypergraph partitioning: applications in vlsi domain. IEEE Trans. Very Large Scale Integr. Syst. 7(1), 69–79 (1999)
32. Kementsietsidis, A., Neven, F., de Craen, D.V., Vansummeren, S.: Scalable multi-query optimization for exploratory queries over federated scientific databases. PVLDB pp. 16–27 (2008)
33. Kotidis, Y., Roussopoulos, N.: Dynamat: A dynamic view management system for data warehouses. In: The ACM Special Interest Group on Management of Data (ACM-SIGMOD). pp. 371–382 (1999)
34. Lanasri, D., Khouri, S., Bellatreche, L.: Trust-aware curation of linked open data logs. In: The INTERNATIONAL CONFERENCE ON CONCEPTUAL MODELING (ER). pp. 604–614 (2020)
35. Larson, P., Yang, H.Z.: Computing queries from derived relations. In: The International Conference on Very Large Data Bases (VLDB). pp. 259–269 (1985)
36. Le, W., Kementsietsidis, A., Duan, S., Li, F.: Scalable multi-query optimization for sparql. In: The International Conference on Data Engineering (ICDE). pp. 666–677 (2012)
37. Liang, X., Elmore, A.J., Krishnan, S.: Opportunistic view materialization with deep reinforcement learning. CoRR abs/1903.01363 (2019), http://arxiv.org/abs/1903.01363
38. Liu, L., Özsu, M.T. (eds.): Encyclopedia of Database Systems, 2nd Edition. Springer (2018)
39. Maier, C., Dash, D., Alagiannis, I., Ailamaki, A., Heinis, T.: PARINDA: an interactive physical designer for postgresql. In: The International Conference on Extending Database Technology (EDBT). pp. 701–704
40. Mami, I., Bellahsene, Z.: A survey of view selection methods. SIGMOD Rec. 41(1), 20–29 (2012)
41. Marroquin, R., Müller, I., Makreshanski, D., Alonso, G.: Pay one, get hundreds for free: Reducing cloud costs through shared query execution. In: ACM Symposium on Cloud Computing. pp. 439–450 (2018)
42. Michiardi, P., Carra, D., Migliorini, S.: Cache-based multi-query optimization for data-intensive scalable computing frameworks. Inf. Syst. Frontiers 23(1), 35–51 (2021)
43. Mistry, H., Roy, P., Sudarshan, S., Ramamritham, K.: Materialized view selection and maintenance using multi-query optimization. In: The ACM Special Interest Group on Management of Data (ACM-SIGMOD). pp. 307–318 (2001)
44. Monika Rokosik, M.W.: Efficient processing of streams of frequent itemset queries. In: The European Conference on Advances in Databases and Information Systems (ADBIS). pp. 15–26 (2014)
45. Mouna, M.C., Bellatreche, L., Narhimene, B.: HYRAQ: optimizing large-scale analytical queries through dynamic hypergraphs. In: IDEAS 2020: 24th International Database Engineering & Applications Symposium, Seoul, Republic of Korea, August 12-14, 2020. pp. 17:1–17:10. ACM (2020), https://dl.acm.org/doi/10.1145/3410566.3410582
46. O'Gorman, K., Agrawal, D., Abbadi, A.E.: Multiple query optimization by cache-aware middleware using query teamwork. In: The International Conference on Data Engineering (ICDE). p. 274 (2002)

47. Pavlo, A., Butrovich, M., Joshi, A., Ma, L., Menon, P., Aken, D.V., Lee, L., Salakhutdinov, R.: External vs. internal: An essay on machine learning agents for autonomous database management systems. IEEE Data Eng. Bull. 42(2), 32–46 (2019)
48. Perez, L.L., Jermaine, C.M.: History-aware query optimization with materialized intermediate views. In: The International Conference on Data Engineering (ICDE). pp. 520–531 (2014)
49. Phan, T., Li, W.: Dynamic materialization of query views for data warehouse workloads. In: The International Conference on Data Engineering (ICDE). pp. 436–445 (2008)
50. Rehrmann, R., Binnig, C., Böhm, A., Kim, K., Lehner, W., Rizk, A.: Oltpshare: The case for sharing in OLTP workloads. Proc. VLDB Endow. 11(12), 1769–1780 (2018)
51. Roukh, A., Bellatreche, L., Bouarar, S., Boukorca, A.: Eco-physic: Eco-physical design initiative for very large databases. Information Systems pp. 44–63 (2017)
52. Roy, P., Sudarshan, S.: Multi-query optimization. In: In [38] (2018), https://doi.org/10.1007/978-1-4614-8265-9_239
53. Savva, F., Anagnostopoulos, C., Triantafillou, P.: Adaptive learning of aggregate analytics under dynamic workloads. Future Gener. Comput. Syst. 109, 317–330 (2020)
54. Scheuermann, P., Shim, J., Vingralek, R.: WATCHMAN : A data warehouse intelligent cache manager. In: The International Conference on Very Large Data Bases (VLDB). pp. 51–62 (1996)
55. Schlag, S.: High-Quality Hypergraph Partitioning. Ph.D. thesis, Karlsruhe Institute of Technology, Germany (2020), https://nbn-resolving.org/urn:nbn:de:101:1-20200304035816201657 65
56. Sellis, T.K.: Multiple-query optimization. ACM Trans. Database Syst. 13(1), 23–52 (1988)
57. Shim, K., Sellis, T.K., Nau, D.S.: Improvements on a heuristic algorithm for multiple-query optimization. Data Knowl. Eng. 12(2), 197–222 (1994)
58. Tapdiya, A., Xue, Y., Fabbri, D.: A comparative analysis of materialized views selection and concurrency control mechanisms in nosql databases. In: IEEE International Conference on Cluster Computing (CLUSTER). pp. 384–388 (2017)
59. Timos K. Sellis, S.G.: On the multiple query optimization problem. IEEE Transactions on Knowledge and Data Engineering pp. 262–266 (1990)
60. Yang, J., Karlapalem, K., Li, Q.: Algorithms for materialized view design in data warehousing environment. In: The International Conference on Very Large Data Bases (VLDB). pp. 136–145 (1997)
61. Yu, J.X., Yao, X., Choi, C.H., Gou, G.: Materialized view selection as constrained evolutionary optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 33(4), 458–467 (2003)
62. Yu, X., Li, G., Chai, C., Tang, N.: Reinforcement learning with tree-lstm for join order selection. In: The International Conference on Data Engineering (ICDE). pp. 1297–1308 (2020)
63. Zdonik, S.B., Maier, D. (eds.): Readings in Object-Oriented Database Systems. Morgan Kaufmann (1990)
64. Zilio, D.C., Rao, J., Lightstone, S., Lohman, G.M., Storm, A.J., Garcia-Arellano, C., Fadden, S.: DB2 design advisor: Integrated automatic physical database design. In: The International Conference on Very Large Data Bases (VLDB). pp. 1087–1097 (2004)
65. Zlatic, V., Ghoshal, G., Caldarelli, G.: Hypergraph topological quantities for tagged social networks. CoRR abs/0905.0976 (2009)

Appendix

```
Q1: select c_nation,s_nation,d_year,sum(lo_revenue) as revenue
from     CUSTOMER, lineorder, SUPPLIER, DATES
where  lo_custkey = c_custkey
and lo_suppkey = s_suppkey
and lo_orderdate = d_datekey
```

```
and c_region = 'AFRICA'
and s_region = 'AFRICA'
and d_year >= 1992 and d_year <= 1997
group by c_nation, s_nation, d_year
order by d_year asc, revenue desc;


Q2: select c_city,s_city,d_year,sum(lo_revenue) as revenue
from     CUSTOMER, lineorder, SUPPLIER, DATES
where lo_custkey = c_custkey
and lo_suppkey = s_suppkey
and lo_orderdate = d_datekey
and c_nation = 'VIETNAM'
and s_nation = 'VIETNAM'
and d_year >= 1993 and d_year <= 1998
group by c_city, s_city, d_year
order by d_year asc, revenue desc;

Q3:select c_city,s_city,d_year,sum(lo_revenue) as revenue
from     CUSTOMER, lineorder, SUPPLIER, DATES
where  lo_custkey = c_custkey
and lo_suppkey = s_suppkey
and lo_orderdate = d_datekey
and (c_city='IRAQ     4' or c_city='JORDAN   6')
and (s_city='IRAQ     4' or s_city='JORDAN   6')
and d_year >= 1992 and d_year <= 1997
group by c_city, s_city, d_year
order by d_year asc, revenue desc;

Q4:select c_city,d_year,sum(lo_revenue) as revenue
from     CUSTOMER, lineorder, DATES
where lo_custkey = c_custkey
and lo_orderdate = d_datekey
and c_nation = 'GERMANY'
and d_year >= 1993 and d_year <= 1998
group by c_city, d_year
order by d_year asc, revenue desc;

Q5: select s_city,p_brand,sum(lo_revenue-lo_supplycost) as profit
from     SUPPLIER,lineorder, PART
where  lo_suppkey = s_suppkey
and lo_PARTkey = p_PARTkey
and s_nation = 'VIETNAM'
and p_category = 'MFGR#34'
group by  s_city, p_brand
order by  s_city, p_brand;

Q6: select c_nation,s_nation,d_year,sum(lo_revenue) as revenue
from     CUSTOMER, lineorder, SUPPLIER, DATES
where lo_custkey = c_custkey
and lo_suppkey = s_suppkey
and lo_orderdate = d_datekey
and c_region = 'AFRICA'
and s_region = 'ASIA'
and d_year >= 1992 and d_year <= 1997
group by c_nation, s_nation, d_year
order by d_year asc, revenue desc;

Q7:select  c_city,d_year,sum(lo_revenue) as revenue
from     CUSTOMER, lineorder, DATES
where  lo_custkey = c_custkey
and lo_orderdate = d_datekey
and c_nation = 'BRAZIL'
and d_year >= 1992 and d_year <= 1997
group by  c_city, d_year
order by  d_year asc, revenue desc;
```

Mustapha Chaba Mouna is a last-year PhD student at University Blida 1, Blida, Algeria. He holds a master's degree in Computer Science from the same university. His research interest focuses on query processing and optimization.

Ladjel Bellatreche is a Full Professor at National Engineering School for Mechanics and Aerotechnics (ISAE-ENSMA), Poitiers, France. He leads the Data and Model Engineering Team of the Laboratory of Computer Science and Automatic Control for Systems (LIAS). He is also a Part Time Professor at Harbin Institute of Technology (HIT) since 2019. His was a visiting professor of the Québec en Outaouais - Canada (2009), a Visiting Researcher at Purdue University - USA (2001) and Hong Kong University of Science and Technology, China (1997-1999). His research interest focuses on Data Management Systems and Semantic Web. He serves as an Associate Editor of the Data & Knowledge (DKE) Journal, Elsevier. His research projects have been funded by 2 EU projects.

Narhimene Boustia is a full Professor at university Blida 1, Blida, Algeria, where she joined as a faculty member since Oct. 2002. Her research interest focuses on data security, access control and data Management. She has co-authored more than 30 papers and received more than 90 citations (H-index=6).

Received: June 06, 2021; Accepted: September 15, 2021.

Computer Science and Information Systems 19(2):763–781 https://doi.org/10.2298/CSIS210820004S

A Neuroevolutionary Method for Knowledge Space Construction

Milan Segedinac[1], Nemanja Milićević[2], Milan Čeliković[1] and Goran Savić[1]

[1] Faculty of Technical Sciences, Trg D. Obradovića 6,
21000 Novi Sad, Serbia
{milansegedinac, milancel, savicg}@uns.ac.rs
[2] SmartCat, Danila Kiša 3V/14,21000 Novi Sad, Serbia
nemanja.milicevic@smartcat.io

Abstract. In this paper we propose a novel method for the construction of knowledge spaces based on neuroevolution. The main advantage of the proposed approach is that it is more suitable for constructing large knowledge spaces than other traditional data-driven methods. The core idea of the method is that if knowledge states are considered as neurons in a neural network, the optimal topology of such a neural network is also the optimal knowledge space. To apply the neuroevolutionary method, a set of analogies between knowledge spaces and neural networks was established and described in this paper. This approach is evaluated in comparison with the minimized and corrected inductive item tree analysis, de facto standard algorithm for the data-driven knowledge space construction, and the comparison confirms the assumptions.

Keywords: Genetic algorithms, Knowledge Space Theory, Neural networks, Educational technology,

1. Introduction

Knowledge Space Theory (KST) gives a theoretical framework for assessing the quality of student's knowledge by representing their knowledge state instead of just quantifying it by giving a numerical grade that would represent the amount of knowledge. Identifying the precise knowledge state is of a key importance since it can direct the forthcoming learning process and suggest which units of knowledge a student should study next.

One of the most important issues in KST [1] is the construction of knowledge spaces, the mathematical models of the structure of students' knowledge [2]. There are two classes of methods that serve this purpose: theory-driven and data-driven. In theory-driven methods, the knowledge space construction is based on the experts' theoretical knowledge about the domain. On the other hand, data-driven methods construct knowledge spaces by analysing students' tests results. Such methods do not require any theoretical assumptions about the domain problems, the relationships among them, nor about the skills that the problems assume. Even though theory-driven techniques are highly useful, they are time consuming and highly labour intensive. To avoid this

disadvantage, the method that we propose in this paper belongs to the family of data-driven knowledge space construction algorithms.

Real educational settings often deal with large and highly interconnected domains [2]. Such domains typically call for large knowledge spaces for representing students' knowledge states. Constructing knowledge spaces for large domains with numerous interconnected problems is a great challenge for data-driven algorithms. The number of potential knowledge spaces grows exponentially as the number of problems in the domain increases. The complexity of this problem is the main motivation for examining the possible alternative methods for knowledge space construction conducted in this research.

The rapid development in the field of Deep Learning in the past decade has called for efficient methods for solving complex optimization problems and has led to the development of new and powerful optimization algorithms. These algorithms can be applied to other fields if analogies between the models in the new fields of application and the originally intended Deep Learning models can be established. The hypothesis of this paper is that, since data-driven knowledge space construction can be observed as a combinatorial optimization problem, optimization techniques developed for the Deep Learning purposes can be directly applied to knowledge space construction as well, if analogies between the original field of application and KST are identified. In that way, we propose a novel method that it is convenient in constructing large knowledge spaces. The method uses a set of analogies between knowledge spaces and neural networks that we establish. The result of this paper is a neuroevolutionary, data-driven method for constructing knowledge spaces.

As such, this method will be useful for both educators and researchers: it will allow educators to utilize KST in teaching subjects with large and complex domains; it will also help educational researchers to study the way students learn such subjects; and the set of analogies between knowledge spaces and neural networks will contribute further development of the field of knowledge space theory by applying other Deep Learning techniques.

The paper consists of 5 sections. Section 2 gives an overview of the related work and the theoretical framework utilized in this paper, namely the Knowledge Space Theory, Neuroevolution, and the NEAT algorithm. In Section 3 we give the description of our method. The evaluation of our method for the knowledge space construction against the most commonly used knowledge space construction algorithm is given in Section 4. The paper concludes with suggestions for future improvement of the proposed method.

2. Related Work

KST [3], [4] is a subfield of mathematical psychology that was initially used mostly for the adaptive assessment of students' knowledge. KST starts from the premise that a domain can be defined as a potentially large but essentially discrete set of units of knowledge, i.e. the problems that students should master. A student that can solve all the problems from a domain is considered to have completely mastered the domain. Most often, a student can solve some, but not all of the problems from the domain. That set of the problems that a student is able to solve is termed as the knowledge state.

The set of all possible knowledge states can be very large. For a domain that consists of 30 problems, there are potentially 2^{30} = 1,073,741,824 knowledge states. But, luckily, in practice most of them are not feasible. For example, a student that cannot find the first derivative of a function will not be able to solve the problem of finding the function minima. In this example, all knowledge sates that would include the latter problem, but would not include the former one would be implausible. Following KST terminology we say that the latter problem surmises the former one.

A knowledge structure consists of a domain together with all feasible knowledge states with requirement that the empty set (representing the student who has just started learning and has not mastered any problem yet) and the domain itself (representing the student who has mastered all the problems from the domain) are feasible knowledge states.

A knowledge structure in which a union of every pair of knowledge states is also a knowledge state (that is closed under union) is termed knowledge space. In such a knowledge structure, if two students were engaged in extensive interactions while studying, it is conceivable that one of them would, at some point in time, acquire the joint knowledge of both [4].

Fig. 1 shows one knowledge space. The nodes in this graph represent the knowledge states and the edges represent surmise relation, which can be defined as follows: problem *a* surmises problem *b*, if from knowing that a student is able to solve the problem *a* we can infer that student is capable of solving problem *b*. In the figure we can see that the domain consists of a set of problems *{a, b, c, d}*. A student can master problems *a* and *d* independently, but in order to master problem *b* they need to be able to solve problem *a*.

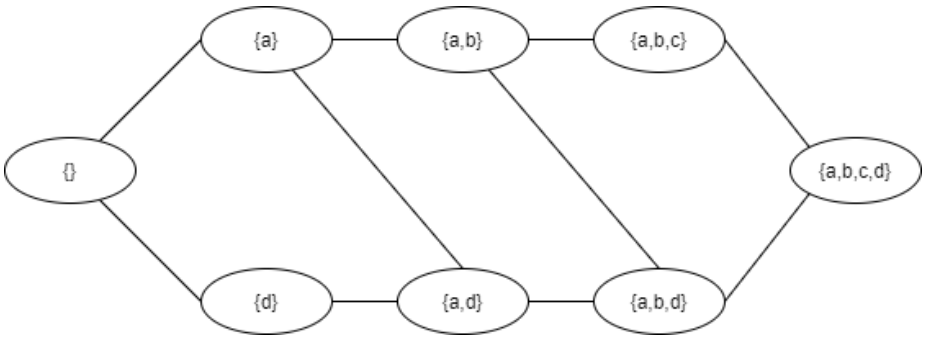


Fig. 1. Knowledge space

Surmise relation can be interpreted so that it gives a set of prerequisites for every knowledge state. Formally, it is defined as the function that associates a family of knowledge states to each knowledge state. The particularly important class of knowledge spaces are those that are closed under intersection. Such knowledge spaces can be represented by a quasi-order, without loss of information [4].

A knowledge space is considered to be a learning space if the following two additional axioms hold:

1. If knowledge state K_2 includes knowledge state K_1, a student can reach K_2 starting from K_1 by mastering one problem at the time.

2. If both K_1 and K_2 are knowledge states for which $K_1 \subset K_2$ holds and if $K_1 \cup \{q\}$ is also a knowledge state where q is an arbitrary problem from domain, then $K_2 \cup \{q\}$ is a knowledge state, as well. This axiom asserts that mastering new problems from a domain will never disable a student from mastering the ones they were able to master before.

2.1. Knowledge Space Construction

There are two categories of the knowledge space construction methods:

1. *Theory driven methods*, that either utilize the experts' knowledge about the domain or rely on the analysis of the problem-solving process, and
2. *Data-driven methods*, based on the analysis of students' response

The methods that utilize experts' knowledge about the domain use specialized algorithms for selecting appropriate combinations of questions and presented them to the experts and construct the knowledge space from the responses. Examples of this approach are the QUERY algorithm [5] and the method proposed by Cosyn and Thiery that combines QUERY algorithm with the analysis of solved tests [6].

Other theory-driven knowledge space construction methods decompose the problems in the domain into sets of motives. An example of such methods is proposed in paper [7]. In this method the information about the motives required for solving the problems is used for constructing the knowledge space.

Another similar theory-driven method observes competencies involved in problem solving process [8]. This method represents competencies by uses revised Bloom's taxonomy for representing the cognitive processes and an ontology for the domain knowledge. Other competence-based methods ([9], [10]) use domain ontology and cognitive process taxonomy to defined the dimensions of the competences and values of these dimensions are interpreted as attributes. The knowledge space constructed by using such methods reflects the assumption that, if there are two problems testing the same domain knowledge, the one that requires a lower cognitive process will precede the one that requires higher cognitive process.

All of these theory-driven methods share the same advantage, that they can be used prior to evaluating students. The most important disadvantage is that the need for continuous involvement of experts in the knowledge space construction process makes them labor intensive. Even though some of them tend to reduce the involvement of experts to some extent (e.g., the one that decomposes the problems in the sets of motives), all of them require a substantial amount of manual work. Another disadvantage of such methods, identified in [11] is that knowledge spaces that experts expect, often do not fit the test data properly so it was concluded that the real knowledge space often differs from the expected one. Theory-driven techniques are also inappropriate for constructing large knowledge spaces.

On the other hand, data-driven methods build a knowledge space starting from a set of student's answers to test questions that can be either right or wrong, i.e. response patterns. Most of these algorithms fall into one of two categories: those that use Boolean analysis to construct the surmise relation and those that construct the knowledge structure directly from the data. The most prominent examples of the first category are Item Tree Analysis (ITA) [12] and Inductive Item Tree Analysis (IITA) [13] while the examples of the second category can be found in the Schrepp's paper [14]. There are

also hybrid methods that combine the data-driven algorithms with skill maps, such as D-SMEP method [15], or with consulting domain experts [6]. All the data-driven algorithms follow a three-step procedure: 1) construct the set of candidate knowledge spaces (2) test the knowledge spaces against a given criterion and (3) choose the best one according to the given criterion. Since all knowledge states are derived from the empirical data, the mentioned methods either consider all the possible knowledge states or impose certain structural constraints to data that allow them to derive the knowledge states that do not occur directly in the students' response patterns. For example, ITA and IITA construct quasi-ordinal knowledge spaces (both closed under union and intersection).

IITA derives a set of quasi-ordinal surmise relations for the given domain and chooses the one that fits the data best. That is achieved by estimating the number of counterexamples for each of them, and the one with minimal discrepancy between the observed and expected number of counterexamples is the fittest [15], [12]. This algorithm has been criticized because of its inductive approach to the construction of the knowledge space; namely is possible the addition of two implications causes an intransitivity if they are added together, but not if added separately [16]. Overcoming this problem by introducing the corrected estimator, and minimizing fit criterion so that it favors quasi-orders that favor smallest minimum discrepancies proposed in paper [16] led to the minimized and corrected IITA, de facto standard algorithm for data drive knowledge space construction. Because of that, the method that we propose is to be compared against the minimized and corrected IITA.

The method proposed in this paper is a purely data-driven one, meaning that it does not require any experts' involvement and results with a knowledge space aligned with the test results. In contrast to inductive data-driven techniques (like IITA), it does not impose any additional restriction to the knowledge space that is constructed, but it can be easily adopted to follow restrictions if needed. In addition, it can be applied to wide domains with large numbers of questions. The method establishes a set of analogies between artificial neural networks and knowledge spaces, and uses these analogies to apply well developed Deep Learning techniques for knowledge space construction. To the best of our knowledge, this approach is novel and there are no other researches that have established such analogies or utilized neuroevolution for the construction of knowledge spaces.

2.2. Neuroevolution

There are topological similarities between feedforward neural networks and knowledge spaces. This fact allows us to observe the problem of knowledge space construction as a special case of the optimization of neural network topology. There are number of ways to optimize the structure of neural networks. One of the most popular is neuroevolution, a family of evolutionary algorithms for the construction and training of neural networks.

Evolutionary Computation. Data driven knowledge space construction requires efficient combinatorial optimization techniques. Evolutionary computation [17], [18] offers a family of such techniques, inspired by the process of biological evolution, that search for suboptimal solutions for a given problem. Such techniques are most often applied in solving optimization problems with large search spaces, that make them particularly interesting candidate for knowledge space construction. In contrast to them, traditional search algorithms choose possible solutions either at random (e.g. random walk algorithm) or by using some heuristics (e.g. gradient descent) and their computational complexity is too high for the mentioned problems.

In order to construct knowledge spaces by applying an evolutionary algorithm it is necessary to choose the appropriate genetic representation, the way in which an individual in the population (i.e. a knowledge space) represents a possible solution of the given problem. That requires representing the set of encoded properties of an individual which forms a genotype. Most often, individuals are represented by the fixed-size sequences of bits (chromosomes), and that will also be the case in this research.

For each individual in the population the fitness function assesses how well it solves the problem. In the case of knowledge spaces, the fitness function tells us how well the knowledge space fits the given test results. The value given by the fitness function is used as a selection criterion when choosing the individuals that are going to be parents for next generation. From the set of parents obtained by selection, reproduction is achieved by applying the crossover operator resulting in the offspring. Just before the new generation is formed, the mutation operator is applied to the individuals that will constitute it. The mutation operator defines small random changes in the chromosome and its goal is to allow the algorithm to check a wider search space, and, in the end, to find a better fitted solution. These steps are repeated until the individual that fulfils the predefined termination condition is met. In this case, that is when the knowledge space that sufficiently fits the assessment results is found.

Neuroevolution. Since evolutionary algorithms are developed for solving complex optimization problems, they can be suitable for construction of knowledge spaces. In that way, the knowledge space construction would start from from a knowledge space and it would proceed by adding new knowledge states and extending the surmise relation throughout the evolution. As knowledge spaces can be observed as feed-forward artificial neural networks there is a possibility of applying a neuroevolutionary algorithm for their construction instead of developing an evolutionary algorithm for that purpose from scratch.

Neuroevolution is a branch of artificial intelligence that uses evolutionary algorithms to construct neural networks – both to generate their topologies and to optimize parameters [19]. The main idea is to iteratively generate, select and cross neural networks until an acceptable suboptimal solution is found, as with the general evolutionary algorithm explained in the previous section. In this process, the neural networks that have better results on the training data and those that generalize the data better are higher ranked by the fitness function.

Neuroevolution gives better results than the traditional methods based on gradient descend in environments with sparse feedbacks, such as training a neural network to win a game that requires a large number of steps to finish. Therefore, neuroevolution can be applied to a broad set of problems where a large set of precisely labelled data is not

available. They impose restrictions that performance can be measured during the training process and that the behaviour of the network can be changed as it evolves.

It should be noted that recent research has shown that simple neuroevolutionary algorithms match the performance of modern sophisticated Deep Learning algorithms based on gradient descent optimization [20]. Some of the most prominent neuroevolutionary algorithms currently are GNARL [21], EPNet [22], NEAT [23], HyperNEAT [24], DXNN [25].

In neuroevolution, there are two ways to map genotype into phenotype: direct and indirect encoding. When direct encoding is applied, the genotype is directly mapped onto the phenotype, meaning that each neuron and each synapse in a neural network has its explicit representation in genotype. In our case, it would mean that every knowledge state and every surmise relation would have their explicit representation in genotype. On the other hand, in indirect encoding, the genotype indirectly specifies how the neural network should be generated. Indirect encoding is often useful for compressing large phenotypes into smaller genotypes, narrowing the search space [24], [26], [27]. In this research we will rely on the direct encoding while indirect encoding will be a topic of our future work.

Neuroevolution was first proposed as an alternative to training neural networks by backpropagation algorithm [28]. Those algorithms were used on networks with fix-topology, meaning that for them only the synaptic weights were subjected to evolutionary optimization, while the topology remained unaltered. Thus, the fix-topology neuroevolution differs from the evolution of the biological neural systems in which the structure of the neural systems itself evolves. One drawback of fixed topology neuroevolution is that in such algorithms the neural network cannot grow through the evolution, and this fact stops the network from becoming able to solve problems harder than the ones it was initially intended for. Fixed topology algorithm would be unsuitable for the construction of knowledge spaces because they will not allow for new knowledge states to be discovered. More recent algorithms solve this issue by evolving the topology of the network together with its synaptic weights.

These algorithms fall into a broad category of topology and weight evolving artificial neural network (TWEANN) algorithms. In them, the topology of the neural network can be changed by adding new neurons and synapses among the neurons with respect to certain constraints. Since the problem that we address in this research requires a topology of the knowledge space to be evolved, it will be solved by an algorithm that belongs to the TWEANN family.

3. Neuroevolutionary Knowledge Space Construction

In this section we present our neuroevolutionary approach to the knowledge space construction. This approach is based on the NEAT algorithm, and for that purpose knowledge spaces are observed as a special kind of neural networks.

3.1. NEAT Algorithm

Stanley and Miikkulainen have shown that simulated evolution of topology together with the synaptic weights give better results than fixed-topology neuroevolution [29], and proposed the NEAT algorithm as a member of TWEANN family. NEAT has solved three problems persistent with the algorithms that have preceded it: (1) giving an adequate genetic representation of the neural networks that enabled meaningful crossing-over of substantially different networks (2) preservation of the topological innovations for a few generations until their characteristics show up properly and (3) minimizing the topology during evolution without additional metrics that would measure the topology complexity. NEAT supports three mutation types: synaptic weight modification, addition of a new synapse and addition of a new neuron that shares the existing synapses. Additionally, NEAT has introduced new cross-over mechanisms and a concept of shared fitness function that increases the diversity.

NEAT uses direct genotype to phenotype encoding, meaning that each neuron and each synapse are explicitly represented. Because of that, there is no need to define new rules for neural network generation from the genotype.

One important feature of the NEAT algorithm is that it gradually evolves small networks, starting from a simple perception to more complex ones that can solve demanding problems. It should be mentioned that the NEAT algorithm has also been modified to be suitable for Deep Learning architectures–CoDeepNEAT [30].

3.2. The Neuroevolutionary Method for Constructing Knowledge Spaces

The NEAT algorithm was initially design to be used for neuroevolution, and in this paper we adapt it so that it can be used for evolutionary construction of knowledge spaces. To achieve this adaptation, we need to define the analogies between knowledge spaces and neural networks. It should be stated that solving problems in other domains by identifying the analogies with artificial neural networks is a well-known approach. In paper [31] neural networks were used for the identification of nonlinear dynamic system by identifying a set of analogies between the artificial neural networks and another nonparametric identification technique. Another example of such an approach is paper [32] where analogies between the inverse problem of choice and neural network learning were utilized. Paper [33] identifies uses the analogies between community structures and neural networks to model the evolution of social networks.

Feedforward neural networks can be considered as directed acyclic graphs where input, hidden and output neurons are vertexes and synapses are edges. Knowledge spaces can also be observed as directed acyclic graphs where knowledge states are vertexes and surmise relations define the edges. In this analogy an empty knowledge state would correspond to an input neuron, the knowledge state consisting of all the items from the domain would be an output neuron, and all the other knowledge states would be hidden neurons, while not every neuron has to be connected to all the neurons in the adjacent layer. The general idea of this analogy is shown in Fig. 2.

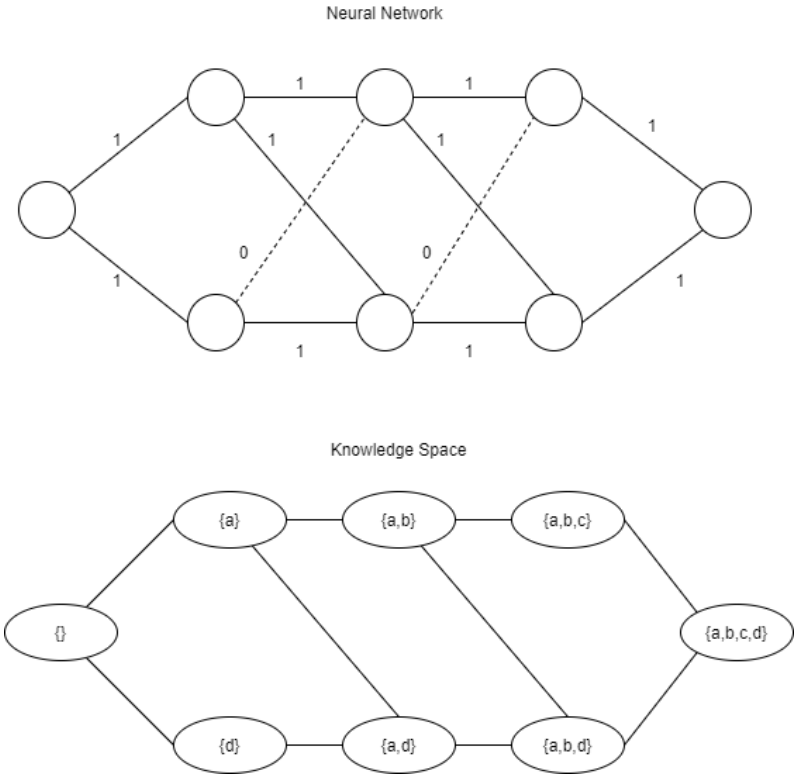


Fig. 2. A feedforward neural network and a knowledge space. The knowledge space can be considered as a neural network with 3 hidden layers and a single neuron in both the input and output layers

The complete set of analogies is given in Table 1.

Table 1. The analogies between neural networks and knowledge spaces

Neural Networks	Knowledge spaces
Neuron	Knowledge state
Neuron in the input layer	Empty knowledge state
Neuron in the output layer	Whole domain
Synapses	Surmise relation
Weights	-
Topology optimization	Knowledge space construction
Training	-

After identification of the analogies, we can observe knowledge spaces as neural networks, we can define the algorithm. It starts with an initial population of knowledge spaces. The initial population consists of knowledge spaces having just one knowledge state – an empty set. The initialization phase is followed by selection, crossing over and

mutation. These steps result with a new population of knowledge spaces. This population is then divided into species and the division is made by the similarity between the individuals. The pseudocode of the algorithm is given in the listing below.

```
population = population_size of empty_knowledge_spaces
for generation_number 0 to number_of_generations
  calculate_fitness for all_knowledge_spaces in population
  if best_fitness > fitness_treshold then
    return knowledge_space_with_best_fitness
  end if
  if (best_fitness_does_not_improve in max_generations) then
    return knowledge_space_with_best_fitness
  end if
  new_population = top_k_best_knowledge_spaces
  while size(new_population) < size(population) do
    parent_1, parent_2 = select_knowledge_space_parents
    child_knowledge_space = crossing_over(parent_1, parent_2)
    child_knowledge_space = mutate(child_knowledge_space)
    new_population.append(child_knowledge_space)
  end while
  population = form_species(new_population)
end for
```

Listing. 1. The algorithm

The remaining section describes the methodology for constructing the knowledge spaces using NEAT algorithm.

Genetic representation. In evolutionary computing, the population consists of individuals represented by genes. In neuroevolution, neural networks are individuals. Since neural network consists of neurons and synapses, the genotype that represents the individual also consists of two types of genes: those that represent neurons and those that represent the synapses. Both of these two types of genes have additional attributes. So, for example, a gene that represents a neuron can also contain the information about the activation function or the type of the neuron (input, hidden or output), while a neuron that represents a synapse can contain the data about the weight, the origin and the destination neurons as well as the indicator if the synapse is active.

It has already been mentioned that, if a knowledge state is observed as a neuron and if surmise relation is observed as a set of synapses, knowledge spaces can be interpreted as a special kind of feed forward neural network. The genome of a knowledge space then has two sets of genes: genes that represent the knowledge states and the ones that represent the surmise relation. It should be noted that not all the information encoded in the genome of a general neural network is required for representing a knowledge space. The genes that represent knowledge states should not contain the information about the activation function and the neuron type.

The surmise relation in a knowledge space represents a discrete phenomenon: either a problem is a prerequisite to the other one, or it is not. Therefore, there is no need for genes that represent the surmise relation to contain the information about the weight. It is enough to represent if a connection exists and which knowledge states are connected with it.

To represent knowledge spaces, two types of genes are required: those that represent knowledge states and those that represent the surmise relation. The method that we propose uses direct genotype to phenotype encoding.

For representing the knowledge states, additional information that does not occur in the original NEAT algorithm should be given. Since a knowledge state is determined by the problems from the domain that the student can solve, the knowledge state is represented by a string of bits where each of them corresponds to one of the problems from the domain. If a student can solve the problem qi, then the i-th bit in the array will have a value of 1. In all other cases, it will have a value of 0. For example, for a domain $Q=\{a,b,c\}$ and a set of knowledge states $\{\{a\},\{b\},\{a,b\},\{a,b,c\}\}$ the bit representation would be: $\{a\}\Rightarrow 100$ $\{b\}\Rightarrow 010$ $\{a,b\}\Rightarrow 110$ $\{a,b,c\}\Rightarrow 111$.

Such a representation has the advantage of using bitwise operations that make the calculation of the fitness function efficient. For example, the fitness function described in this paper uses symmetric distance between a knowledge state and a response pattern. That can be achieved by subtraction and conjunction.

Fitness function. Choosing the right fitness function is very important for the convergence of the algorithm. The fitness function should measure how well an individual solves the initial optimization problem. In this approach, the candidate knowledge spaces are being generated through crossover and mutation, and for each of them the fitness function should evaluate how well the obtained knowledge space is aligned with the dataset. In addition to that, the fitness function should be quick to evaluate, because it is being evaluated a large number of times.

Because of these reasons, the discrepancy measure described in the paper [34], and based upon the k-modes algorithm is used as a fitness function. For the sake of the comprehensibility of the paper, the description of this discrepancy measure given in the paper [34] follows.

In this method the dataset is a collection of response patterns, each of which represents the problems from the domain that the student has solved correctly. The observed dataset is represented by the pair $(\mathscr{R}, F)$, where $\mathscr{R}\subseteq 2^Q$ is a subset of the partition set of domain Q and $F:\mathscr{R}\rightarrow\mathbb{R}$ is a function that assigns the number of occurrences to each response pattern. For function F, it holds $F(R)\geq 0$ for every $R\in\mathscr{R}$ and $\Sigma F(R)=N$, for $R\in\mathscr{R}$ and for N students' responses.

For every knowledge structure $\mathcal{K}$ over the domain Q, partitioning N response patterns into $|\mathcal{K}|$ classes is represented by partition function $f:\mathscr{R}\times\mathcal{K}\rightarrow\mathbb{R}$ that fulfils two conditions:

1. $f(R,K)\geq 0$ for $R\in\mathscr{R}$ and $K\in\mathcal{K}$,
2. $\Sigma f(R,K)=F(R)$, for $K\in\mathcal{K}$ and $R\in\mathscr{R}$.

The partition function can be interpreted in the following manner: for given $K\in\mathcal{K}$ and $R\in\mathscr{R}$ the function assigns $f(R,K)$ out of $f(R)$ pattern response R occurrences to the class that is being represented by the knowledge state K. The second condition guaranties that each R is going to be assigned to some class represented by the knowledge state K. For all possible partition functions for $\mathscr{R}$ and $\mathcal{K}$, the one that minimizes a certain dissimilarity measure is to be chosen. One such simple measure between $K\in\mathcal{K}$ and $R\in\mathscr{R}$ is the cardinality of their symmetric distance, $d(R,K)=|(K\setminus R)\cup(R\setminus K)|$.

Dissimilarity measure within a class given by the knowledge state $K \in \mathcal{K}$ is the weighted sum of the symmetric distances: $Df(\mathcal{R},) = \Sigma f(R,K) \cdot d(R,K)$, $R \in \mathcal{R}$.

And the total dissimilarity of a knowledge structure $\mathcal{K}$and a dataset $(\mathcal{R},F)$ is the sum of all dissimilarity measures within all the classes: $Df(\mathcal{R},\mathcal{K}) = \Sigma\Sigma f(R,K) \cdot d(R,K) R \in \mathcal{R} K \in \mathcal{K}$.

The value of the fitness function for an individual in the population (that is a candidate knowledge space) is obtained from the dissimilarity Df that tells us how much the knowledge space differs from the dataset.

Selection and crossover operators. The proposed method uses the selection and crossover operators as the original the NEAT algorithm. In contrast to traditional genetic algorithms, NEAT divides the population into species. For each species, the total fitness function is calculated as the average fitness of all the individuals inside it. Then to each species a number of offspring is assigned and it is proportional to the value of the species' fitness function. The total number of offspring has to be equal to the predefined number of individuals in the population. That means that the species that have a greater fitness value will give more offspring in the next generation and that good solutions will propagate to next generations. After the numbers of offspring for the species are determined, from each species two parents are chosen to produce the offspring. After the crossover, the obtained individual is subjected to the mutation operator described in the following section.

Mutation. In each step, chosen knowledge spaces mutate and a new knowledge state can be added to them. This new knowledge state that is added to a knowledge space selected from the population differs from a knowledge state that already exists in the knowledge space for a single problem from the domain. That way, it is guaranteed that the knowledge space is well-graded meaning that it is also a learning space.

Every genome goes through the mutation phase in which there is a fixed probability that a mutation will occur. In this phase a knowledge state is chosen at random from the knowledge space that this genome represents and, if the mutation occurs, the resulting knowledge state includes one more problem from the domain. It is possible that such a knowledge state already exists in the knowledge space represented by the selected individual. In that case, the knowledge space is not changed by the mutation. In the other case, if the new knowledge state is added to the knowledge space, the surmise relation is being updated as well.

Since the mutation is random, it can happen that the newly obtained knowledge space has a smaller fitness value than the original one, before the mutation. In that case, the resulting knowledge space will have a smaller chance of survival and reproduction. All the knowledge states have the same probability of being chosen for the mutation.

Speciation. Speciation is an idea from the original NEAT algorithm that relies on the fact that new evolutions do not show their strengths right away, but need a couple of generations to emerge. Traditional genetic algorithms put new structures at a disadvantage. NEAT solves this problem by grouping individuals into species, and keeping them protected inside their species until they are fully developed.

The species are formed in accordance to the similarity of the individuals inside it. For that purpose, the function that measures the similarity between two genomes and the threshold that determines if the two individuals should be considered to belong to the same species are defined. The measure of genetic similarity is defined as:

$$\delta = \frac{c_1 E}{N} + \frac{c_2 D}{N} + c_3 S. \quad (1)$$

where E is the number of excesses, D is the number of disjoint genes and S is the sum of the symmetric gene distances. Coefficients c_1, c_2 and c_3 serve as weights that represent the importance of the addends. N is the total number of genes and it serves for normalization. Speciation is applied in this research in the same way as in original NEAT algorithm because it can turn out that the evolution of the knowledge space will not show its strengths right away, but it might turn out to show them in future.

4. Evaluation

In this section we present the evaluation of the proposed method. As explained above, today's most widely used data-driven technique for knowledge space construction is minimized and corrected IITA. Hence, the method that we propose is compared against that algorithm

The algorithms were compared with respect to their ability to reconstruct knowledge spaces from the dataset. In order to determine metrices the following was used:

1. The number of knowledge states $|\mathcal{K}e|$
2. The true positive rate (TPR), that is the percentage of knowledge states from the original knowledge space identified in the constructed knowledge space,

$$TPR = \frac{|\mathcal{K}_e \cap \mathcal{K}|}{|\mathcal{K}|}, \quad (2)$$

3. The false positive rate (FPR), that is the percentage of the knowledge states that exist in the constructed knowledge space that do not exist in the original knowledge space

$$FPR = \frac{|\mathcal{K}_e \backslash \mathcal{K}|}{|\mathcal{K}_e|}, \quad (3)$$

4. The discrepancy measure described previously in this paper

$$\boldsymbol{D(\mathcal{R}, \mathcal{K}_e)} \quad (6)$$

The dataset, consisting of response patterns used for the evaluation, is generated with three variable parameters: the number of problems in the domain, the number of knowledge states and the number of responses in the dataset. All the datasets are generated by using basic local independent model (BLIM), a probabilistic model of knowledge structures [35].

4.1. The dataset

BLIM defines the relation between a response pattern R and a knowledge state K with the following equation:

$$P(R) = \sum_{K \in \mathcal{K}} P(R \mid K) \cdot \pi_K, \tag{4}$$

In this equation *P(R)* is the probability of choosing a student with a response pattern *R*, $P(\mathcal{R} \mid \mathcal{K})$ is the conditional probability of responding with the pattern R for the given knowledge state *K*, and π_K is the probability of a student having a knowledge state *K*. Respecting the assumption that domain problems are locally independent in respect to the given knowledge states, for any response pattern *R* and knowledge state *K*, the conditional probability $P(\mathcal{R} \mid \mathcal{K})$ is given in the following equation:

$$r(R,K) = \left[\prod_{q \in K \setminus R} \beta_q\right]\left[\prod_{q \in K \cap R} (1-\beta_q)\right]\left[\prod_{q \in R \setminus K} \eta_q\right]\left[\prod_{q \in \overline{R \cup K}} (1-\eta_q)\right] \tag{5}$$

Here, $\beta_q, \eta_q \in [0,1]$ are, respectively, the probabilities of a careless mistake and a lucky guess.

In order to use BLIM to simulate the response patterns for N students, we must have the knowledge structure. The first step is to take the knowledge state *K* with the given probability. Then, for each problem in the domain $q \in Q$, random careless errors and lucky guesses are formed with the probabilities β_q and η_q. For simulating the datasets in this paper, the following parameters have been varied:

- The number of problems in the domain $q \in Q$
- The number of knowledge states $K \in \mathcal{K}$
- The number of response patterns *N*

The values of the parameters for the datasets simulations used in this paper are given in the table 2.

The appropriate number of knowledge states for the given domain depends on the number of problems in the domain. The larger the domain is, the more knowledge states there are. So, in this simulation, there were 30 or 60 knowledge states for the domains with 10 problems and 100 knowledge states for the domain with 15 problems.

Table 2. the values of the parameters used for simulating the datasets

Combination number	$\|Q\|$	$\|\mathcal{K}\|$	N
1	10	30	250
2	10	60	500
3	10	30	250
4	10	60	500
5	15	100	1000

Larger domains require more response patterns in order to construct a knowledge space. For the knowledge spaces with 30 or 60 knowledge states there were 250 or 500 response patterns. For the one with 100 knowledge states, there were 1000 response patterns.

Three knowledge spaces were constructed at random for the values in the table: $\mathcal{K}1$ for combinations 1 and 2; $\mathcal{K}2$ for combinations 3 and 4; and $\mathcal{K}3$ for combination 5. For each combination, 10 simulated datasets with N response patterns were generated using BLIM. Parameters β_q and η_q are taken with uniform probability distribution from interval (0,0.05]. The probabilities $\pi_{\mathcal{K}}$ are taken from the interval [0.4, 0.6] also with uniform distribution, and, afterwards normalized so to equal 1. For each combination, 10 datasets were generated making 50 datasets in total.

4.2. The Comparison

This section describes the comparison of the proposed method and minimized and corrected IITA. For each of the generated datasets the learning space was constructed by using these two algorithms. For the proposed method, when there are 10 problems in the domain the population consisted of 1024 individuals, and, for 15 problems in the domain there were 2048 individuals in the population. The number of generations was 100, but early stoppage was allowed if there were no significant improvements in 20 generations. These four metrices were measured for all 10 simulated datasets for each combination. Afterwards the mean was taken to represent the performance of the algorithm for the combination

4.3. The Results

This section gives the results of the evaluation of the proposed method. In order to position it relative to the current state in the field, we have compared these results with the ones obtained by state of the art minimized and corrected IITA which is de facto standard for data-driven knowledge space construction. Table 2 shows the metrices of the neuroevolutionary method in parallel to the minimized and corrected IITA.

Table 3. The results

	Neuroevolutionary method				Minimized and corrected IITA			
	$\lvert\mathcal{K}_e\rvert$	TPR	FPR	$D(\mathcal{R},\mathcal{K}_e)$	$\lvert\mathcal{K}_e\rvert$	TPR	FPR	$D(\mathcal{R},\mathcal{K}_e)$
1	31.20 (1.78)	0.97 (0.04)	0.07 (0.02)	23.20 (31.05)	26.70 (4.12)	0.85 (0.11)	0.04 (0.03)	245.80 (208.02)
2	31.30 (1.79)	0.97 (0.03)	0.08 (0.03)	395.20 (346.45)	29.10 (3.86)	0.90 (0.08)	0.07 (0.05)	673.50 (570.02)
3	57.10 (3.33)	0.90 (0.05)	0.05 (0.04)	48.70 (44.73)	49.60 (10.06)	0.83 (0.17)	0.00 (0.00)	138.00 (185.93)
4	64.3 (2.19)	0.99 (0.01)	0.07 (0.04)	94.90 (23.65)	53.10 (3.33)	0.89 (0.06)	0.00 (0.00)	451.90 (208.75)
5	113.9 (7.19)	0.98 (0.01)	0.13 (0.05)	143.00 (60.52)	63.4 (4.54)	0.63 (0.05)	0.00 (0.00)	3678.60 (351.46)

Neuroevolutionary method results in knowledge spaces of the similar size to the original ones. This is the consequence of the appropriate fitness function which penalizes large knowledge spaces. The results show that our method resulted in slightly larger knowledge spaces than the minimized and corrected IITA algorithm.

The neuroevolutionary method gives good results of TPR since it manages to find almost all the knowledge states from the original knowledge space. It outperformed minimized and corrected IITA and it is particularly noticeable for large learning spaces. For combination 5 with 100 knowledge states the neuroevolutionary method has identified 98% of the knowledge states, while the other algorithm managed to find 63% of them.

On the other hand, minimized and corrected IITA proved to be slightly better in the case of FPR. In the mentioned case of 100 knowledge states the neuroevolutionary method had a FPR of 0.13 while this value for the other algorithm was 0. One of the reasons for this is the fact that the search space in the case of this knowledge space is much larger. Having larger populations might result in a better FPR rate, and it will be a subject of future research. We can also see that the neuroevolutionary method had a smaller discrepancy measure than IITA.

5. Conclusion

We propose in the paper a novel method for data-driven knowledge space construction. The proposed method is based upon the neuroevolutionary computing, which is one of the contributions of this paper. To the best of our knowledge, there are no similar attempts in related works. The main motivation for this approach results from the fact that neuroevolution allows solving complex optimization problems, and, therefore, the proposed method is appropriate for the construction of knowledge spaces for large and highly interconnected domains.

The method was based on the NEAT algorithm. For that purpose, a set of analogies between neural networks and knowledge spaces was proposed. The identification of these analogies itself is also a contribution of this paper, because it allows a wide range of Deep Learning techniques to be applied in the field of Knowledge Space Theory, not just to the construction of knowledge spaces.

In order to apply neuroevolution to the problem of knowledge space construction, we have proposed a genetic representation of the knowledge space, introduced the fitness function for knowledge spaces in accordance with the response patterns, and defined the speciation operator for knowledge spaces. To the best of our knowledge, these problems were not priorly solved.

The neuroevolutionary method has been compared with minimized and corrected IITA which is de facto standard data-driven knowledge space construction algorithm. From this evaluation we can conclude that the neuroevolutionary method is capable of constructing knowledge spaces from the students' response patterns. As expected, the evaluation suggests that it is more appropriate method for constructing large knowledge spaces than minimized and corrected IITA. There is still lot of space for research concerning this algorithm. First of all, optimizing fitness function and population size might result in a better FPR without compromising TPR, and this is one of the topics for future research. Secondly, we can see that both the neuroevolutionary method and minimized and corrected IITA have their strengths. Therefore, future research should combine these two algorithms to harvest the benefits of both. In addition, the proposed method can be combined with theory-driven techniques that will yield the initial knowledge spaces before assessing knowledge, and applying the neuroevolutionary method to refine them afterwards in our future work.

The method is useful for educators and education researchers. To the educators, it will allow KST to be utilized in teaching subjects with large and complex domains. To the educational researchers, it will help studying the way students learn such subjects. In addition to that, the identification of the set of analogies between knowledge spaces and neural networks will contribute further development of the field of Knowledge Space Theory by allowing the application of other Deep Learning techniques.

References

1. Doignon, J.-P., Falmagne, J.-C.: Spaces for the assessment of knowledge. International journal of man-machine studies, Vol. 23, No. 2, 175–196. (1985)
2. Ünlü, A., Sargin, A.: DAKS: an R package for data analysis methods in knowledge space theory. Journal of Statistical Software, Vol. 37, No. 1, 1-31. (2010) 3. Doignon, J.-P., Falmagne, J.-C.: Knowledge spaces, Springer Science & Business Media, (2012)
4. Falmagne, J.-C., Doignon, J.-P.: Learning spaces: Interdisciplinary applied mathematics, Springer Science & Business Media, (2010)
5. Koppen, M.: Extracting human expertise for constructing knowledge space: an algorithm. Journal of mathematical psychology, Vol. 37, No. 1, 1–20. (1993)
6. Cosyn, E., Thiéry, N.: A practical procedure to build a knowledge structure. Journal of mathematical psychology, Vol. 44, No 3, 383–407. (2000)
7. Schrepp, M., Held, T., Albert, D.: Component-based Construction of Surmise Relations for Chess Problems. In D. Albert & J. Lukas (Eds.), Knowledge Spaces: Theories, Empirical Research, and Applications (pp. 41–66). Mahwah: NJ. (1999)

8. Marte, B., Steiner, C. M., Heller, J., Albert, D.: Activity and Taxonomy-Based Knowledge Representation Framework. International Journal of Knowledge and Learning, Vol. 4, No. 1, 189–202. (2008)
9. Albert, D., Held T.: Establishing knowledge spaces by systematical problem construction. In D. Albert (Ed.), Knowledge Structures. New York: Springer Verlag, 78–112. (1994)
10 Albert, D., Held, T.: Component based knowledge spaces in problem solving and inductive reasoning, In D. Albert & J. Lukas (Eds.), Knowledge Spaces: Theories, Empirical Research, and Applications. Mahwah, NJ: Lawrence Erlbaum Associates., 15–40. (1999)
11. Segedinac, M., Horvat, S., Rodić, D., Rončević, T., Savić, G.: Using knowledge space theory to compare expected and real knowledge spaces in learning stoichiometry, Chemistry Education Research and Practice (CERP), Vol. 19, No 3, 670-680. (2018)
12. Ünlü, A., Albert, D.: The correlational agreement coefficient ca (≤, d)—a mathematical analysis of a descriptive goodness-of-fit measure. Mathematical Social Sciences, Vol. 48, No. 3, 281–314. (2004)
13. Schrepp, M.: A method for the analysis of hierarchical dependencies between items of a questionnaire. Methods of Psychological Research Online, Vol. 19, No.1, 43–79. (2003)
14. Schrepp, M.: Extracting knowledge structures from observed data. British Journal of Mathematical and Statistical Psychology, Vol. 52, No. 2, 213–224. (1999)
15. Spoto,A., Stefanutti, L., Vidotto, G.: An iterative procedure for extracting skill maps from data. Behavior research methods, Vol. 48, No. 1, 729–741, (2016)
16. Sargin, A., Ünlü, A.: Inductive item tree analysis: Corrections, improvements, and comparisons. Mathematical Social Sciences, Vol. 58, No 3, 376-392. (2009)
17. Rechenberg, I.: Evolution strategy: Optimization of technical systems by means of biological evolution. Fromman-Holzboog: Stuttgart, Vol. 104, No 1, 15–16. (1973)
18. Holland, J. H., Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press. (1992)
19. Stanley, K. O.: Neuroevolution: A different kind of deep learning. (2017) [Online]. Available: https://www. oreilly. com/ideas/neuroevolution-a-different-kind-of-deep-learning. (current December 2020)
20. Such, F. P., Madhavan,,V., Conti, E., Lehman, J., Stanley, K. O., Clune, J.: Deep neuroevolution: Genetic algorithms are a competitive alternative for training deep neural networks for reinforcement learning. arXiv (2017) [Online]. Available: https://arxiv.org/abs/1712.06567 (current December 2020)
21. Angeline, P. J., Saunders. G. M., Pollack, J. B.: An evolutionary algorithm that constructs recurrent neural networks. IEEE transactions on Neural Networks, Vol. 5, No. 1, 54–65. (1994)
22. Yao, X., Liu, Y.: A new evolutionary system for evolving artificial neural networks. IEEE transactions on Neural Networks, Vol. 8, No. 3. 694–713. (1997)
23. Stanley, K. O., Miikkulainen, R.: Efficient evolution of neural network topologies. In Proceedings to CEC'02, Honolulu, HI, USA, USA. (2002)
24. Gauci, J., Stanley, K.: Generating large-scale neural networks through discovering geometric regularities. In Proceedings to GECCO '07, London, England. (2007)
25. Sher, G. I.: Handbook of neuroevolution through Erlang. Springer Science & Business Media. (2012)
26. Gruau, F.: Neural network synthesis using cellular encoding and the genetic algorithm. LIP-IMAG. (1994)
27. Clune, J., Stanley, K. O., Pennock, R. T., Ofria, C.: On the performance of indirect encoding across the continuum of regularity. IEEE Transactions on Evolutionary Computation, Vol. 15, No. 3., 346–367. (2011)
28. Rumelhart, D. E., Hinton, G. E., Williams, R. J.: Learning internal representations by error propagation., ICS, San Diego, CA, USA. (1985)

29. Stanley, K. O., & Miikkulainen, R.: Efficient evolution of neural network topologies. In Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (2002).
30. Miikkulainen, R., Liang, J., Meyerson, E., Rawal, A., Fink, D., Francon, O., Raju, B., Shahrzad, H., Navruzyan, A., Duffy, N.: Evolving deep neural networks. In Kozma, R., Alippi, C., Choe, Y., Morabito, F. C. (Eds.) Artificial Intelligence in the Age of Neural Networks and Brain Computing. Elsevier, 293–312. (2019)
31. Masri, S. F., Chassiakos, A. G., Caughey, T. K.: Identification of nonlinear dynamic systems using neural networks. Journal of Applied Mechanics, Vol 60, No 1, 123-133. (1993)
32. Mikoni, S. V.: Neural network approach to the formation models of multiattribute utility. International Journal Information Models & Analyses, Vol 3, No 1, 3-9. (2014)
33. Rituraj, K, Biswal, B.: A model for evolution of overlapping community networks. Physica A: Statistical Mechanics and its Applications, Vol 474, No 1, 380-390. (2017)
34. de Chiusole, D., Stefanutti, L., Spoto, A.: A class of k-modes algorithms for extracting knowledge structures from data. Behavior research methods, Vol 49, No 4, 1212-1226. (2017)
35. de Chiusole, D., Stefanutti, L., Anselmi, P., Robusto, E.: Assessing parameter invariance in the BLIM: Bipartition models. Psychometrika, Vol. 78, No.4, 710–724. (2013)

Milan Segedinac received his M.Sc. degree in 2008 and Ph.D. in 2014 in Computer Science from the University of Novi Sad, Faculty of Technical Sciences. He holds an associate professor position at the same faculty. He has authored papers in international and national journals and conferences in the field of computer enhanced education.

Nemanja Milićević has received his M.Sc. (2019) degree from the Faculty of Technical Sciences at the University of Novi Sad. He is currently a data scientist at SmartCat. His research interests are in the field of deep learning and AI supported education.

Milan Čeliković received his M.Sc. degree from the Faculty of Technical Sciences, at University of Novi Sad in 2009. He received his Ph.D. degree in 2018, at the University of Novi Sad, Faculty of Technical Sciences. Currently, he works as an assistant professor at the Faculty of Technical Sciences at the University of Novi Sad, where he lectures several Computer Science and Informatics courses. His main research interests are focused on: Databases, Database management systems, Information Systems and Software Engineering.

Goran Savić is an associate professor within the Department of Computing and Control, Faculty of Technical Sciences, University of Novi Sad. He received his M.Sc. degree in 2006 and Ph.D. degree in 2011, all in Computer Science from the University of Novi Sad, Faculty of Technical Sciences. His research interests are e-learning and enterprise information systems.

Received: August 20, 2021; Accepted: January 25, 2022.

Computer Science and Information Systems 19(2):783–801 https://doi.org/10.2298/CSIS210830005Z

Hyper-graph Regularized Subspace Clustering With Skip Connections for Band Selection of Hyperspectral Image

Meng Zeng[1], Bin Ning[1⋆], Qiong Gu[1], Chunyang Hu[1], and Shuijia Li[2]

[1] School of Computer Engineering, Hubei University of Arts and Science
Xiangyang, Hubei, China, 441053
zengmeng@cug.edu.cn, {ningbin2000, qionggu, huchunyang}@hbuas.edu.cn
[2] School of Computer Science, China University of Geosciences
Wuhan, Hubei, China, 430074
shuijiali@cug.edu.cn

Abstract. The Hughes phenomenon of Hyperspectral images (HSIs) with the hundreds of continuous narrow bands makes the computational cost of HSIs processing high. Band selection is an effective way to solve such a problem and a lot of band selection methods have been proposed in recent years. In this paper, a novel hyper-graph regularized subspace clustering with skip connections (HRSC-SC) is proposed for band selection of hyperspectral image, which is a clustering-based band selection method. The networks combine subspace clustering into the convolutional auto-encoder by thinking of it as a self-expressive layer. To make full use of the historical feature maps obtained from the networks and tackle the problem of gradient vanishing caused by multiple nonlinear transformations, the symmetrical skip connections are added to the networks to pass image details from encoder to decoder. Furthermore, the hyper-graph regularization is presented to consider the manifold structure reflecting geometric information within data, which accurately describes the multivariate relationship between data points and makes the results of clustering more accurate so that select the most representative band subset. The proposed HRSC-SC band selection method is compared with the existing robust band selection algorithms on Indian Pines, Salinas-A, and Pavia University HSIs, showing that the results of the proposed method outperform the current state-of-the-art band selection methods. Especially, the overall accuracy of the clustering is the best on three real HSIs compared to other methods when the band selection number is 25, reaching 82.62%, 92.48%, and 96,5% respectively.

Keywords: Band selection, hyper-graph regularization, skip connections, subspace clustering, hyperspectral image

1. Introduction

Hyperspectral images (HSIs) with hundreds of narrow bands containing abundant spatial and spectral information so that it can identify the region of interest. Due to the high redundancy, the so-called Hughes phenomenon happens on HSIs frequently and increases computation complexity. Band selection (BS) is an effective way to reduce the dimensionality of HSIs, aiming to select the significant bands with the most information from the original data set as a band subset. The details of band selection are shown in Fig. 1. The

⋆ Corresponding author

BS methods do not destroy the physical properties of the HSIs, which is different from the feature extraction methods that transform the physical characteristics of HSIs. Therefore, the BS methods are easier to explain than the feature extraction methods. BS methods can be classed as supervised and unsupervised fashions [1]. The supervised methods have to apply the prior knowledge, and the unsupervised methods are the opposite. Considering the difficulty to get the labeled samples of HSIs in reality, the unsupervised methods have better application prospects and attracted more attention in recent years.

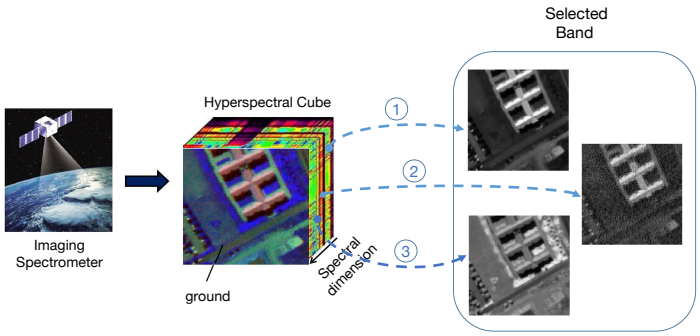


Fig. 1. Details of band selection.

The unsupervised BS methods can be divided into three categories: the searching-based methods, the ranking-based methods, and the clustering-based methods. The searching-based methods, such as multi-objective optimization based band selection (MOBS) [2], optimize the given metric using the heuristic searching, but such methods cost much time on heuristic searching. The ranking-based methods select band according to the importance of the bands, which allocate a rank for each band, such as maximum-variance principal component analysis (MVPCA) [3], sparsity-based band selection (SpaBS) [4], and Laplacian score (Lap-score)[5]. The clustering-based methods cluster the bands into several categories according to the assumption that all bands can be separated based on the similarity of the bands, and we can select the band subset from the categories, i.e., sparse non-negative matrix factorization clustering (SNMF) [6], improved sparse subspace clustering (ISSC) [7], deep subspace clustering for Band Selection (DSCBS) [8], etc. The clustering-based methods make full consideration the interaction between bands so that the clustering-based methods get great success over the past years. In this paper, we use a cluster-based method for band selection, which is based on the deep subspace clustering (DSC) [9] network. The proposed method makes full use of the globally nonlinear spectral-spatial relationship, improving the performance of band selection.

However, although DSC has a good effect on general images, it is difficult to achieve better results for complex HSIs. For example, the average clustering accuracy of DSC can reach 98% on the ORL data set [9], but only 75% on the Indian Pines data set and 70% on the Pavia University data set [10]. Therefore, skip connections and hyper-graph learning are introduced to optimize the network and improve the accuracy of BS.

The skip connections [11], also known as residual connections, can skip one or more layers in different layers to build extra connections between nodes. These skip connections can solve the problem of gradient vanishing, recover the original image and make full use of the historical feature maps obtained from the networks by pass image details from the encoder to the decoder in auto-encoder networks. The graph learning [12,13] is an important topic and has been widely used in image processing, which reflects the geometric information of data by learning the manifold structure. However, the simple graph model can only represent the simple relationships between data so it can not obtain robust results in complex high-order relationships of images. To overcome the obstacles, hyper-graph learning [14] has been introduced to describe the multivariate relationship between simples of complex HSIs. To better compare the pros and cons of different methods, we summarize the above methods, as shown in table 2.

In this paper, a novel hyper-graph regularized subspace clustering with skip connections (HRSC-SC) for band selection of hyperspectral image is introduced, which is the clustering-based band selection method. The subspace clustering is used to combine the convolutional auto-encoder by thinking of it as a self-expressive layer. The symmetrical skip connections are added to the convolutional auto-encoder (CAE) [15] for HSIs clustering, which pass image details from encoder to decoder. In addition, the hyper-graph regularized is introduced to describe the multivariate relationship between simples of complex HSIs. The main contributions of this paper are summarized as follows:

1. We propose a hyper-graph regularized subspace clustering with skip connections (HRSC-SC) for band selection of the hyperspectral image.
2. The symmetrical skip connections are added into the convolutional auto-encoder with a self-expressive layer to make full use of the historical feature maps obtained from the networks and tackle the problem of gradient vanishing caused by multiple nonlinear transformations, which pass image details from encoder to decoder and produce more beneficial representation for better clustering.
3. The hyper-graph regularized is introduced to describe the multivariate relationship between simples of complex HSIs that can fully consider the manifold structure reflecting geometric information within data, making the modeling of images more accurate. Three HSI data sets are utilized to evaluate the performance and efficiency of the proposed band selection algorithm. Experimental results show that the proposed HRSC-SC band selection method has state-of-the-art performance, outperforming the current robust band selection methods.

To make the subsequent expression clearer, we construct table 1 to summarize the acronyms in this paper. The rest of the paper is structured as follows. In Section 2, we briefly review the sparse subspace clustering and the convolutional auto-encoder. Then, the details of the proposed HRSC-SC for band selection are introduced in section 3. In section 4, we evaluate the HRSC-SC for three well-known HSI data sets. Finally, conclude with a summary and discuss the future research directions in section 5.

2. Previous Work

2.1. Sparse Subspace Clustering

The subspace clustering methods consist of two steps: first, evaluate the affinity for each pair of simples to build an affinity matrix, which is the most crucial step and determines

Table 1. Acronyms in the paper.

Full name	Acronyms
Average Accuracy	AA
Alternating Direction Method of Multipliers	ADMM
Auto-Encoders	AE
Band Selection	BS
Convolutional Auto-Encoder	CAE
Deep Subspace Clustering	DSC
Deep Subspace Clustering for Band Selection	DSCBS
Hyper-graph Regularized Subspace Clustering with Skip Connections	HRSC-SC
Hyperspectral Images	HSIs
Improved Sparse Subspace Clustering	ISSC
Laplacian score	Lap-score
Multi-objective Optimization based Band Selection	MOBS
Maximum-variance Principal Component Analysis	MVPCA
Overall Accuracy	OA
Self-expressive	SE
Sparse Non-negative Matrix Factorization	SNMF
Sparsity-based Band Selection	Spa-BS
Sparse Subspace Clustering	SSC

Table 2. Comparison of band selection methods.

Methods	Based	Technology	Band interaction	Spectral-spatial
MOBS [2]	searching	heuristic searching	no	not consider
MVPCA [3]	ranking	maximum-variance	no	not consider
SpaBS [4]	ranking	sparse representation	no	not consider
Lap-score [5]	ranking	Laplacian score	no	not consider
SNMF [6]	clustering	sparse non-negative matrix factorization	yes	not consider
ISSC [7]	clustering	sparse subspace clustering	yes	not consider
DSCBS [8]	clustering	deep auto-encoder+subspace clustering	yes	consider
HRSC-SC	clustering	Hyper-graph+Skip Connections+DSC	yes	full consider

the results of clustering; second, utilize spectral clustering [16] or normalized cuts [17] method by using the affinity matrix. Here, we briefly introduce the SSC [18] method. Let the data set be the size of $M \times N$, where M and N denote the dimension of the feature and the number of data points respectively. Assuming that all the data points lie in a union of t affine subspaces $S = S_1, S_2, \cdots S_t$, where the t subspace has dimensions $\{d_i\}_{i=1}^{t}$ and $d_1 + d_2 + \cdots d_t = M$. Based on the hypothesis above, the optimization equation can be represented as:

$$\min_{\mathbf{C}} \|\mathbf{C}\|_1 , s.t. \mathbf{Y} = \mathbf{YC} + \mathbf{N}, diag\,(\mathbf{C}) = 0 \tag{1}$$

where $\mathbf{C}$ is the coefficient matrix. $\mathbf{Y}$ represents the data points. $\mathbf{N}$ denotes the error matrix. 1 stands for the l_1-norm regularization, which can be 0, 1, 2 in other subspace clustering methods. To avoid trivial solution, the $diag\,(\mathbf{C})$ is constrained to be 0. The ADMM [19,20] method is applied to optimize Eq.1. Then, the similarity graph is constructed by

the coefficient matrix $\mathbf{C}$, the affinity matrix $\mathbf{M}$ can be written as:

$$\mathbf{M} = |\mathbf{C}| + |\mathbf{C}|^T \tag{2}$$

In the end, according to the similarity graph, the results of the clustering can be obtained by the spectral clustering method.

2.2. Convolutional Auto-encoder

The AE consists of the symmetrical encoder and decoder structure, which can convert the data points into the latent space representation. In recent years, the AE methods have been widely used and achieved state-of-the-art performance in learning data deep representation, e.g. variational auto-encoder [21], sparse auto-encoder [22], and denoising auto-encoder [23].

The structure of the CAE is similar to the AE, which consists of the symmetrical convolutional and deconvolutional layers. Here, we can define the convolutional layer as $\varphi = E(x; \alpha_e)$, in which φ denotes the latent representation (or bottleneck) to reveal the intrinsic information of the input data, x and α_e stand for data points and parameters respectively. Analogously, the deconvolutional can be defined as $\hat{x} = D(\varphi; \alpha_d)$, where $\hat{x}$ is the reconstruction of input data and α_d represents the parameters of the decoder. Then, we can define the loss function as:

$$\mathcal{L}(\alpha_e; \alpha_d) = \frac{1}{2} \sum_{i=1}^{N} \|x_i - \hat{x}_i\|_F^2 \tag{3}$$

3. Proposed Method

3.1. Convolutional Auto-encoder with Self-expressive layer

The main structure of the HRSC-SC is to insert a SE layer into a Convolutional CAE [23] to learn a representation for HSI. The SE layer between the encoder and decoder of the CAE to imitate the "self-expressive" property of the traditional subspace clustering. Therefore, the SE layer is as similar as SSC [18], which is defined as:

$$\hat{\mathbf{Z}} = \mathbf{C}\mathbf{Z} \tag{4}$$

where $\hat{\mathbf{Z}}$ denotes the reconstruction of $\mathbf{Z}$, which is the input of the SE layer. $\mathbf{C}$ is the coefficient matrix obtained from the SE layer. The loss function of the SE layer as follows:

$$\mathcal{L}_{SE} = \frac{\lambda_1}{2} \left\| \mathbf{Z} - \hat{\mathbf{Z}} \right\|_F^2 + \frac{\lambda_2}{2} \|\mathbf{C}\|_F^2 \tag{5}$$

There are two parts in Eq.5: the first part is the reconstruction error; the second part uses an F-norm regularization to constrain $\mathbf{C}$. Here, we have not to use the condition of $diag(\mathbf{C}) = 0$ like the traditional SSC because using F-norm without diagonal constraints will not lead to trivial solutions [24].

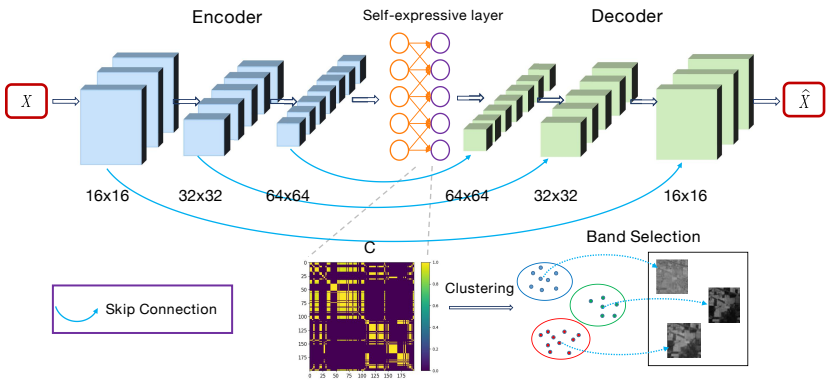


Fig. 2. Overall architecture of the HRSC-SC.

As shown in Fig. 2, giving the input HSI $\mathbf{X} = \{x_i\}_{i=1}^{m}$ and it can be coded through the encoder to obtain the latent representations $\mathbf{Z}$, the function of the encoder is defined as $\mathbf{Z} = f_\beta(\mathbf{X})$. Symmetrically, the refactored $\mathbf{X}$ can be defined as $\hat{\mathbf{X}} = g_\gamma(\mathbf{Z})$. Each layer of the CAE followed by a batch normalization and a ReLU activation. The loss function of the CAE and the CAE with SE layer are defined as Eq.6 and Eq.7, respectively:

$$\mathcal{L}_{AE} = \frac{1}{2}\left\|\mathbf{X} - \hat{\mathbf{X}}\right\|_F^2 \tag{6}$$

$$\mathcal{L}_{AES}(\beta;\gamma) = \mathcal{L}_{AE} + \mathcal{L}_{SE} \tag{7}$$

3.2. Skip Connections

Using the CAE with SE layer for HSI clustering can solve the problem of the nonlinear subspaces. However, all nodes in the SE layer are connected by linear weights that have no bias and non-linear activations so that N^2 parameters exist in the SE layer, making the SE layer the focus of training and overwhelming the optimize of CAE networks. Besides, the multiple layers networks may cause the problem of vanishing gradient.

To deal with the problems above, the symmetrical skip connections are introduced into the CAE. As shown in Fig. 2, skip shortcuts connect the convolution feature map and their deconvolution feature map in a symmetrical manner [25]. The gradients can back-propagate to the corresponding encoder layers directly without going through the SE layer. Ideally, using skip connections can make the network's train from scratch easier. Suppose x_j represents the j-th encoder layer mapping and $\tilde{x_j}$ is the corresponding j-th underlying decoder mapping, the skip connection mapping can fit:

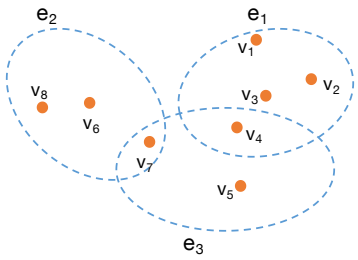
$$\mathcal{H}(x_j) = \tilde{x_j} - x_j \tag{8}$$

According to Eq.8, the j-th underlying decoder mapping can be:

$$\tilde{x_j} = \mathcal{H}(x_j) + x_j \tag{9}$$

3.3. Hyper-graph Learning

According to the graph learning methods [12], the potential geometry of high dimensional data points can be retained by the neighbor graph of the original data [13]. Therefore, we can use the theory to consider the manifold structure reflecting geometric information within data. Different from the traditional graph learning that can only explain the simple relationships between data points, the hyper-graph explains the complex relationships by connecting three or more vertices to describe the multivariate relationships of data accurately, achieving outstanding performance in recent works[26]. Therefore, we introduce the hyper-graph regularized into the subspace clustering networks to fully consider the multivariate relationships of complex HSIs. The hyper-graph structure can be represented as Fig. 3.



(a)

	e1	e2	e3
v1	1	0	0
v2	1	0	0
v3	1	0	0
v4	1	0	1
v5	0	0	1
v6	0	1	0
v7	0	1	1
v8	0	1	0

(b)

Fig. 3. (a) Illustration of the hyper-graph. (b) Corresponding to (a), $\mathbf{V} = \{v_i\}_{i=1}^{m}$ is a vertex and $\mathbf{E} = \{e_j\}_{j=1}^{n}$ denote a hyper-edge, set $(v_i, e_j) = 1$ if the vertex i on the hyper-edge or $(v_i, e_j) = 0$ when the vertex i is not on the hyper-edge.

Assuming there is a hyper-graph $\mathbf{G} = (\mathbf{V}, \mathbf{E}, \mathbf{W})$, in which $\mathbf{V} = \{v_i | i = 1, 2, \cdots, m\}$ denotes the multiple non-empty vertex set, $\mathbf{E} = \{e_j | j = 1, 2, \cdots n\}$ is non-empty hyper-edge subsets and $\mathbf{W}$ represents the weight matrix. We can define the $w\,(e)$ as weight of each hyper-edge. The incident matrix ζ is:

$$\zeta\,(v, e) = \begin{cases} 1, if\ v \in e \\ 0, if\ v \notin e \end{cases} \tag{10}$$

Then, the degree of a vertex v and the hyper-graph e can be defined as Eq.11 and Eq.12 respectively:

$$d\,(v) = \sum_{\{e \in E | v \in V\}} w\,(e) = \sum_{e \in E} w\,(e)\,\zeta\,(v, e) \tag{11}$$

$$\phi(e) = |e| = \sum_{v \in V} \zeta(v, e) \tag{12}$$

Finally, the hyper-graph Laplacian matrix $\mathbf{L}_h$ can be written as:

$$\mathbf{L}_h = \mathbf{D}_v - \zeta \mathbf{W} \mathbf{D}_e^{-1} \zeta^T \tag{13}$$

where D_v denotes the degree matrix of vertex and D_e is the degree matrix of hyper-edge, both of them are the diagonal matrices.

3.4. Training and Band Selection

Algorithm 1 Pseudocode of HRSCNet

Input: image data set $\mathbf{X}$; number of clusters: k; hyper-parameters: λ_1, λ_2, λ_3.
Output: Clustering results.

```
Preprocess image data set;
Initialize C, β and γ of the HRSCNet in Eq.14 ;
Compute the hyper-graph Laplacian matrix L_h according
  to Eq.13;
while maximum iteration is met do
    Calculate the output of the encoder: Z = f_β(X);
    Calculate the output of the SE layer: Ẑ = CZ;
    Calculate the output of the decoder: X̂ = g_γ(Z);
    Calculate the loss according to Eq. 14;
    Update C, β and γ using Adam optimizer;
end
Construct affinity matrix according to Eq.15;
Calculate clustering results using spectral clustering;
Return clustering results.
```

There are four parts included in the HRSC-SC: the CAE, the SE layer, the skip connections, and the hyper-graph regularization. Therefore, the loss function of the HRSC-SC contains the CAE loss L_{AE} in Eq.6, the SE layer loss L_{SE} in Eq.5, and the hyper-graph regularized loss $\mathcal{L}_h = \frac{\lambda_3}{2} Tr\left(\mathbf{C}^T \mathbf{L}_h \mathbf{C}\right)$, expressed as:

$$\begin{aligned} \mathcal{L}(\mathbf{C}; \beta; \gamma) &= \mathcal{L}_{AE} + \mathcal{L}_{SE} + \mathcal{L}_h \\ &= \frac{1}{2} \left\| \mathbf{X} - \hat{\mathbf{X}} \right\|_F^2 + \frac{\lambda_1}{2} \left\| \mathbf{Z} - \hat{\mathbf{Z}} \right\|_F^2 + \frac{\lambda_2}{2} \|\mathbf{C}\|_F^2 + \frac{\lambda_3}{2} Tr\left(\mathbf{C}^T \mathbf{L}_h \mathbf{C}\right) \end{aligned} \tag{14}$$

where λ_1, λ_2 and λ_3 in $\mathcal{L}_{SE}$ and $\mathcal{L}_h$ denote the balancing parameters. The Adam [20] gradient descent method can be utilized to optimize Eq.14. It is worth noticing that we can train the HRSC-SC from scratch because of the skip connections technique, which is different from some existing deep subspace clustering methods that require pre-training.

According to the Fig. 2, we can get the coefficient matrix $\mathbf{C}$ after training the networks. Then, using $\mathbf{C}$ to construct a symmetric matrix $\mathbf{M}$:

$$\mathbf{M} = |\mathbf{C}| + |\mathbf{C}|^T \tag{15}$$

It can be seen in Fig. 2, the subspace clustering method is used to get the clustering results by cluster $\mathbf{M}$ into k classes. Then, the average of bands in each class is deemed as a cluster center, and we calculate the distance from each band to the cluster center. Finally, the selected band is the band closest to the cluster center in each category.

To express the HRSC-SC more clearly, the pseudocode of the proposed algorithm is shown in Algorithm 1.

4. Results

4.1. Data Set and Experimental Settings

We utilize three widely used images data sets for experiments: Indian Pines, Salinas-A, and Pavia University data sets [3]. For the convenience of experiment, the subscence located are used on Indian Pines and Pavia University data sets at $[50 \sim 120, 50 \sim 120]$ and $[200 \sim 300, 100 \sim 200]$, respectively. The details of the three data sets are shown in Table 3.

Table 3. Summary of Indian Pines, Salinas-A and Pavia University data sets.

Data sets	Indian Pines	Salinas-A	Pavia University
Pixels	70×70	86×83	100×100
Bands	200	204	103
Sensor	AVIRIS	AVIRIS	ROSIS

In the HRSC-SC network, the layers of encoder and decoder are all set to 3 and the encoder and the decoder have a symmetrical structure. Therefore, if the channels of the encoder in three layers are set to 16, 32, 64, respectively, the channels of the decoder will be set to 64, 32, 16. The learning rate is set to 1.0×10^{-4} and the hyper-parameters λ_1, λ_2 and λ_3 are all set to 1.0. For a more intuitive description, the hyper-parameters setting of all comparison BS methods is shown in Table 4. All methods except the MOBS method run in Python 3.6 and the MOBS method run in Matlab 2016. All code of the methods is run on the Intel Core i5 3.10GHz.

4.2. Comparison of Performance

In this experiment, the number of the selected bands is changed in the range of 5 to 30, and the interval is set as 5. The SVM classifier [27,28] is used for all band selection methods to evaluate the OA, AA, and kappa coefficient (Kappa) of the different methods. To ensure the fairness of the experiment, we select 5% of labeled samples from each data set as the training set, and others as the testing set. To better test the performance of the algorithm, we evaluate all methods for 10 independent rums. The seven well-known band selection methods, Lap-score [5], SpaBS [4], ISSC [7], SNMF [6], MVPCA [3], DSCBS [8] and MOBS [2], are used as comparison methods to compare with the proposed HRSC-SC method on Indian Pines, Salinas-A, and Pavia University data sets.

[3] http://www.ehu.eus/ccwintco/index.php?title=Hyperspectral_Remote_Sensing_Scenes

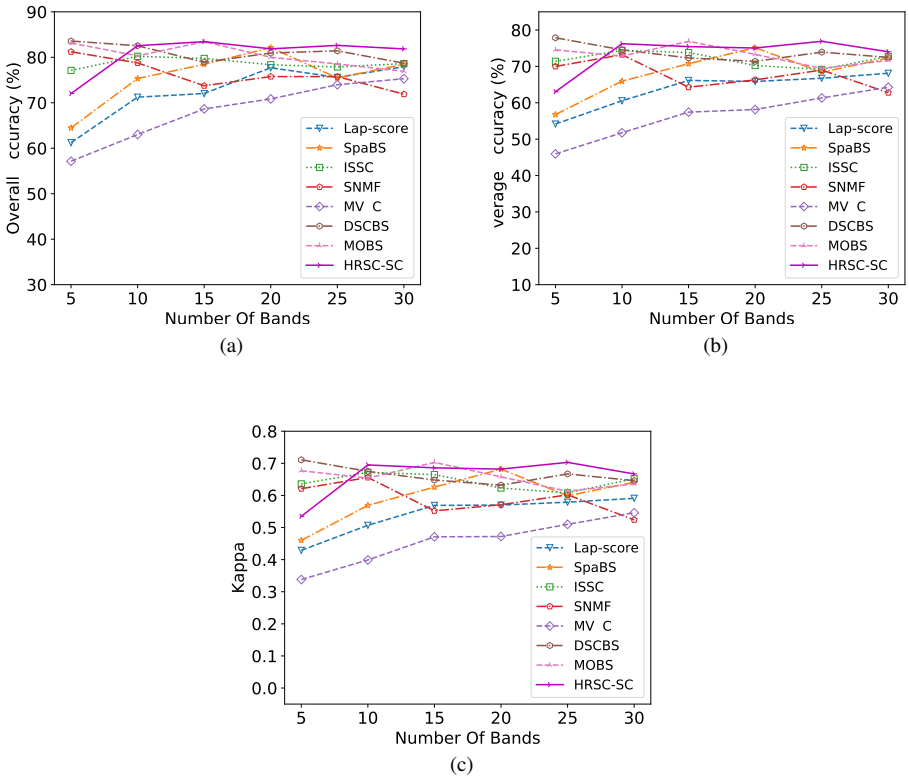


Fig. 4. Performance comparison of different BS methods with different band subset sizes on Indian Pines data set:(a) OA, (b) AA, and (c) Kappa.

The performance comparison of different BS methods with different band subset sizes is shown in Fig. 4 to Fig 6. Overall, it can be seen that the proposed HRSC-SC method achieves the best band selection results on three data sets. As shown in Fig. 4, the OA, AA, and Kappa of HRSC-SC are lower than some methods, but it achieves the best performance than other sever methods when the selected bands are 10 to 30. In addition, the HRSC-SC, SpaBS, ISSC, and MOBS show the best performance when selecting 15 bands, and show the downward trend after 15, that is the accuracy increases first and decreases with the selected bands increasing. The above phenomenon is called Hughes, which is mentioned in section 1. However, the HRSC-SC also achieves the best results than others, which shows the superiority of the HRSC-SC. It is noticed that the ranking-based methods, i.e. MVPCA, SpaBS, and Lap-score, achieve relatively poor accuracy than other approaches because of the increased chances of the misestimating band due to the noise of HSI.

In Fig. 5, similar to Indian Pines, the HRSC-SC shows the best performance when selecting more than 10 bands. Different from Indian Pines, the HRSC-SC has no obvious Hughes phenomenon on Salinas-A. But other methods, like SpaBS, ISSC, and MOBS,

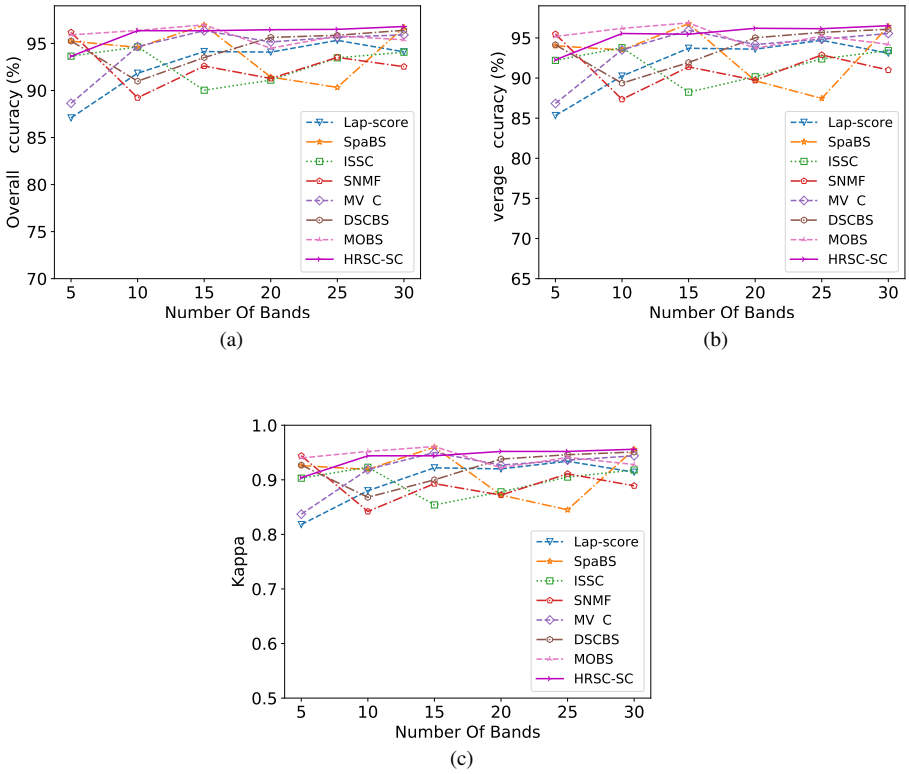


Fig. 5. Performance comparison of different BS methods with different band subset sizes on Salinas-A data set:(a) OA, (b) AA, and (c) Kappa.

also have the Hughes phenomenon. Therefore, HRSC-SC achieves significant results and outperforms all other methods.

In Fig. 6, the HRSC-SC shows great performance when the selected bands are 5 to 15. Then, the results of HRSC-SC are slightly inferior to the DSCBS method. However, the overall performance comparison result is similar to Indian Pines and Salinas-A data sets. The clustering-based methods can fully consider the mutual relations between different bands, especially the proposed method, which added the skip connections and the hyper-graph regularization to consider the deeper relationship of data. Consequently, the proposed clustering-based method can achieve better results than other methods.

4.3. Analysis of the Selected Bands

We can analyze the selected bands according to the results of the tables and figures. To ensure the results more obvious, 15 bands are selected for all selected bands experiments on three data sets. Table 5 shows the selected bands of all methods when the number of selected bands is 15 on three data sets. Relatively, Fig. 7, Fig. 8, and Fig. 9 show the

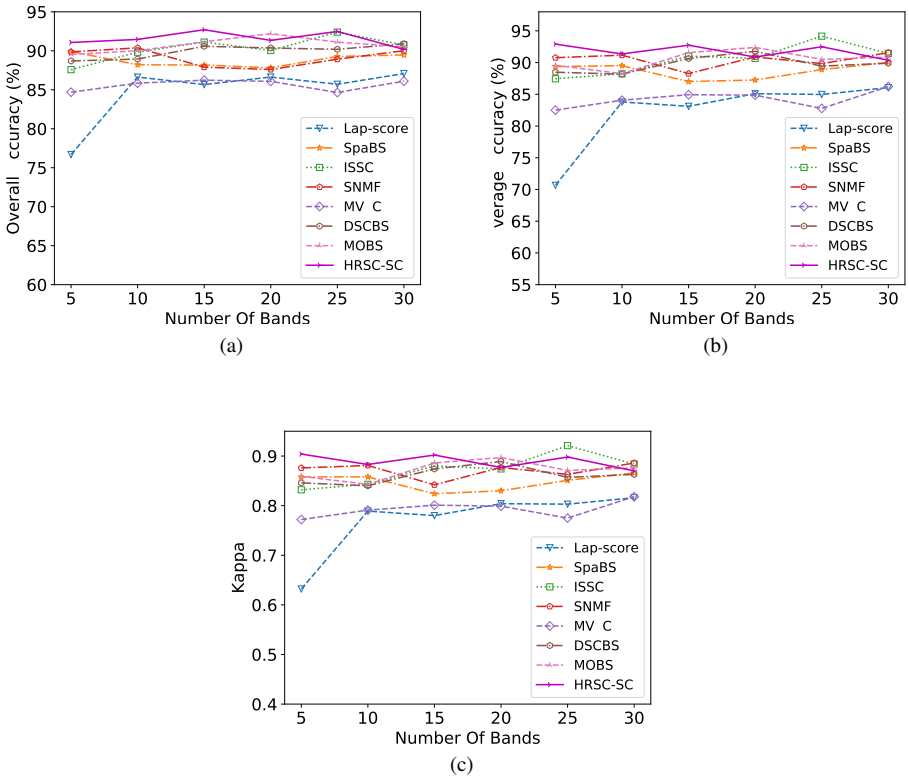


Fig. 6. Performance comparison of different BS methods with different band subset sizes on Pavia University data set:(a) OA, (b) AA, and (c) Kappa.

locations of the selected bands (above) on the spectrum, and the entropy curve (below) of three data sets.

In Fig. 7, the value of entropy is relatively average. For HRSC-SC, the selected bands have uniform distribution and no continuous bands. As for other algorithms, they all have continuous bands, especially MVPCA and Lap-score. The selected bands of the proposed HRSC-SC methods are most evenly distributed.

In Fig. 8, the value of entropy is similar to Indian Pines, it can be seen that the entropy curve has a relatively uniform distribution, which illustrates that better results can be obtained when the band selection is more uniform. In contrast, the MVPCA and the Lap-score methods select a lot of continuous bands. The other comparison algorithms select the relatively evenly bands. However, all of them select some continuous bands, which affect the final classification results. Therefore, the HRSC-SC method achieves the best performance.

Fig. 9 shows the selected bands and the entropy curve of Pavia University. As shown in this figure, the value of entropy is smooth initially and gradually increases, and reaches stability in the end. According to the distribution of entropy, choose server bands in num-

Table 4. Hyper-parameters setting of all comparison BS methods.

Methods	Hyper-paramenters
Lap-score	-
SpaBS	$\lambda = 1e2$
ISSC	$\lambda = 1e5$
SNMF	$maxiter = 100$
MVPCA	-
DSCBS	$\alpha = 1.0, \lambda = 1e-3$
MOBS	$maxiter = 100, NP = 100$
HRSC-SC	$\lambda_1 = \lambda_2 = \lambda_3 = 1.0$

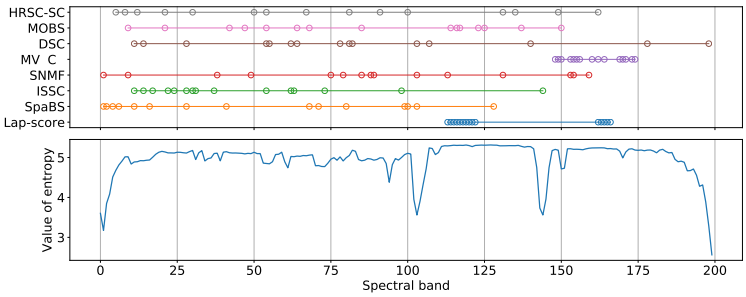


Fig. 7. The best 15 bands of Indian Pines data set selected by different BS methods (above) and the entropy value of each band (below).

bers 0 to 40, and approximately the average selection of bands after number 40 can get good results. Compare with the algorithm, HRSC-SC selects server bands and approximately average selects bands after band number 40. The performance of other methods is the same as using Indian Pines and Salinas-A. Therefore, HRSC-SC can get the best results for band selection on three data sets.

4.4. Impact of Epochs

As shown in Fig. 10, we plot the accuracy of the HRSC-SC and the training losses on different epochs to analyze the convergence on three data sets. The selected bands are set to 20. In Fig. 10 (a), we set the number of iterations from 0 to 200. It can be seen that the lower accuracy at the beginning, and the accuracy increases to maximum when the epoch increases close to 100, then fluctuate within a certain range the loss stabilizes. We set the number of iterations from 0 to 100 on the Salinas-A data set. According to the Fig.10 (b), the accuracy increases with the losses decrease, and then the accuracy reaches the maximum value when the epoch increases close to 50. When the epoch increases close to 100, the accuracy fluctuates within a certain range and the loss tends to be 0 while the accuracy achieve stable. As for Pavia University, we set the number of iterations from 0 to 300. As shown in Fig. 10 (c), before the number of iterations is 100, the accuracy fluctuates greatly. When it is close to 150 times, it tends to stabilize and reach a larger value, and the loss tends to be 0. Combining the above results, to get the best accuracy

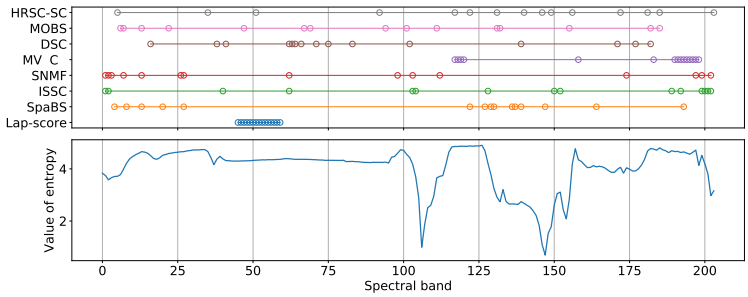


Fig. 8. The best 15 bands of Salinas-A data set selected by different BS methods (above) and the entropy value of each band (below).

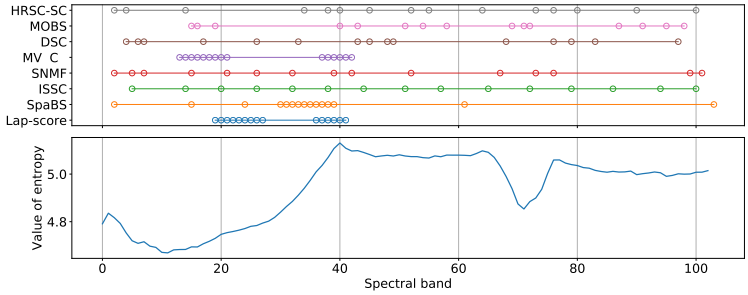


Fig. 9. The best 15 bands of Pavia University data set selected by different BS methods (above) and the entropy value of each band (below).

and cost minimal time, the number of the iterations for Indian Pines, Salinas-A, and Pavia University data sets are set to 100, 50, and 150, respectively.

4.5. Visualization of Affinity Matrix

The affinity matrices of three data set is shown in Fig. 11. According to the theory of the affinity matrix in section 3, the affinity matrix has a symmetrical and sparse block diagonal structure. The block of the affinity matrix can be used to cluster, which is the crucial element for subspace clustering. As shown in Fig. 11, the block is independent relatively and the affinity matrix obtained from the three data sets has good symmetry so that the affinity matrices achieve robust performance.

5. Conclusions

In this paper, a hyper-graph regularized subspace clustering with skip connections (HRSC-SC) was proposed for band selection of the hyperspectral image. The networks combine subspace clustering into the convolutional auto-encoder by thinking of it as a self-expressive layer. Moreover, the symmetrical skip connections are added to the networks

Table 5. The best 15 bands of three data sets (Indian Pines, Salinas-A, Pavia University) selected by different band selection methods.

Data sets	Methods	Selected bands
Indian Pines	Lap-score	[45,46,47,48,49,50,51,52,53,54,55,56,57,58,59]
	SpaBS	[4,8,13,20,27,122,127,129,130,136,137,139,147,164,193]
	ISSC	[1,2,40,62,103,104,128,150,152,189,192,199,200,201,202]
	SNMF	[1,2,3,7,13,26,27,62,98,103,112,174,197,199,202]
	MVPCA	[117,118,119,120,158,183,190,191,192,193,194,195,196,197,198]
	DSCBS	[16,38,41,62,63,64,66,71,75,83,102,139,171,177,182]
	MOBS	[6,7,13,22,47,67,69,94,101,111,131,132,155,182,185]
	HRSC-SC	[5,35,51,92,117,122,131,140,146,149,156,172,181,185,203]
Salinas-A	Lap-score	[45,46,47,48,49,50,51,52,53,54,55,56,57,58,59]
	SpaBS	[4,8,13,20,27,122,127,129,130,136,137,139,147,164,193]
	ISSC	[1,2,40,62,103,104,128,150,152,189,192,199,200,201,202]
	SNMF	[1,2,3,7,13,26,27,62,98,103,112,174,197,199,202]
	MVPCA	[117,118,119,120,158,183,190,191,192,193,194,195,196,197,198]
	DSCBS	[16,38,41,62,63,64,66,71,75,83,102,139,171,177,182]
	MOBS	[6,7,13,22,47,67,69,94,101,111,131,132,155,182,185]
	HRSC-SC	[5,35,51,92,117,122,131,140,146,149,156,172,181,185,203]
Pavia University	Lap-score	[19,20,21,22,23,24,25,26,27,36,37,38,39,40,41]
	SpaBS	[2,15,24,30,31,32,33,34,35,36,37,38,39,61,103]
	ISSC	[5,14,20,26,32,38,44,51,57,65,72,79,86,94,100]
	SNMF	[2,5,7,15,21,26,32,39,42,52,67,73,76,99,101]
	MVPCA	[13,14,15,16,17,18,19,20,21,37,38,39,40,41,42]
	DSCBS	[4,6,7,17,26,33,43,45,48,49,68,76,79,83,97]
	MOBS	[15,16,19,40,43,51,54,58,69,71,72,87,91,95,98]
	HRSC-SC	[2,4,14,34,38,40,45,52,55,64,73,76,80,90,100]

to pass image details from encoder to decoder, which can make full use of the historical feature maps obtained from the networks and tackle the problem of gradient vanishing caused by multiple nonlinear transformations. Besides, we introduce the hyper-graph regularized to consider the manifold structure reflecting geometric information within data to accurately describe the multivariate relationship between data points and make the results of HSIs clustering more accurate. We execute the experiments on three HSI data sets for the proposed HRSC-SC method to show that the proposed method has state-of-the-art performance.

However, there are still the following problems that need to be improved in future work. First, due to the training characteristics of deep learning, the running time of the proposed HRSC-SC method spends much, which is an important topic to research in future works. Second, the proposed HRSC-SC method shows good performance for the band selection of HSIs, which can be used to study the clustering of HSIs in the future. Third, due to the limitation of training time and equipment, we only intercepted part of the three datasets for experiments. In future work, we will try to use the entire dataset for testing to verify the effectiveness of the proposed algorithm. Finally, a self-supervised network structure is used for learning in this paper, in future work, other methods, such as generative adversarial networks, will be introduced to optimize the network structure.

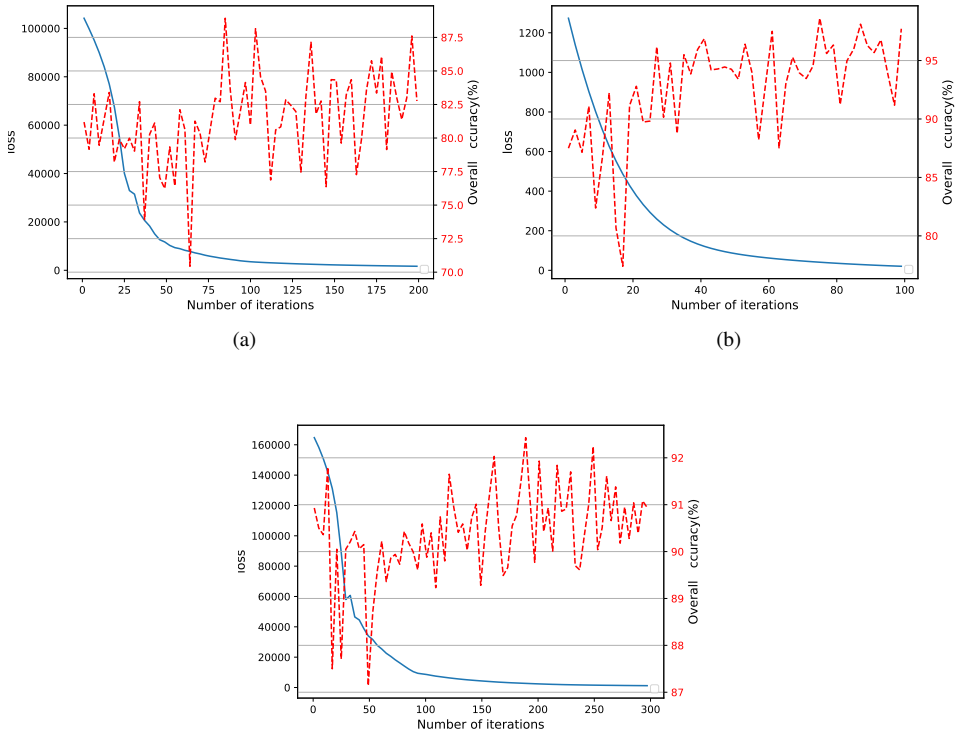


Fig. 10. Loss curve (blue full line) and accuracy curve (red dashed line) for the proposed HRSC-SC approach on three data sets. (a) Indian Pines, (b) Salinas-A, and (c) Pavia University.

Acknowledgments. This work was supported in part by the Guidance Programs of Science and Technology Funds of the Xiangyang city under Grant 2020ZD32, in part by the Major Research Development Program of Hubei Province under Grant 2020BBB092. (Corresponding author: B. Ning)

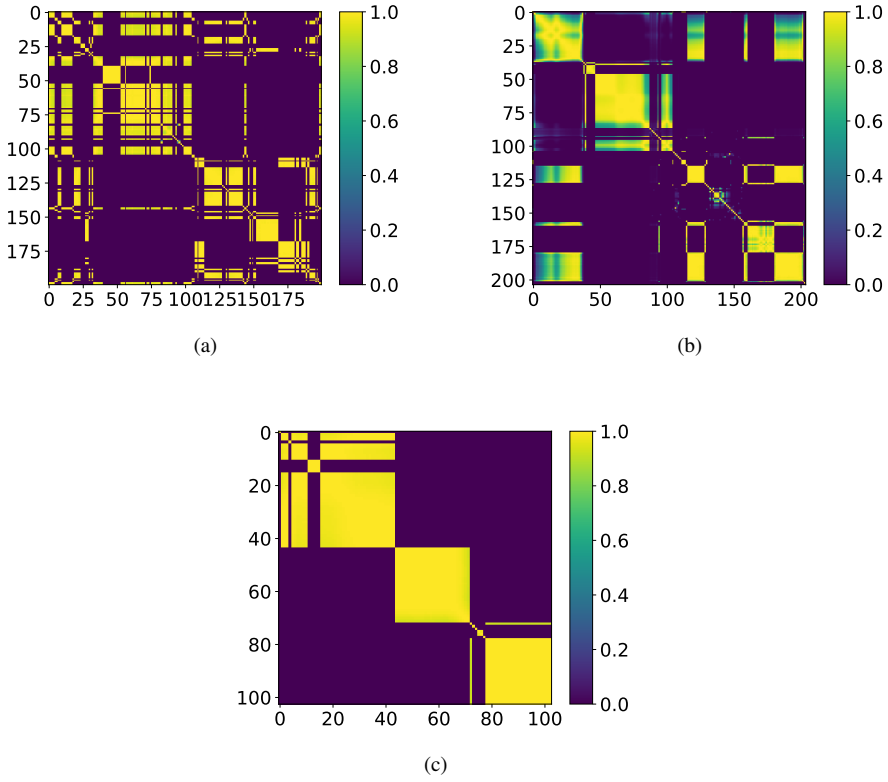


Fig. 11. Visualization of Affinity Matrix on three data sets. (a) Indian Pines, (b) Salinas-A, and (c) Pavia University.

References

1. Mateus Habermann, Vincent Fremont, and Elcio Shiguemori. Unsupervised hyperspectral band selection using clustering and single-layer neural network. *Revue Francaise de Photogrammetrie et de Teledetection*, 2018, 09 2018.
2. P. Hu, X. Liu, Y. Cai, and Z. Cai. Band selection of hyperspectral images using multiobjective optimization-based sparse self-representation. *IEEE Geoscience and Remote Sensing Letters*, 16(3):452–456, 2019.
3. Chein-I Chang, Qian Du, Tzu-Lung Sun, and M. L. G. Althouse. A joint band prioritization and band-decorrelation approach to band selection for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 37(6):2631–2641, 1999.
4. K. Sun, X. Geng, and L. Ji. A new sparsity-based band selection method for target detection of hyperspectral image. *IEEE Geoscience and Remote Sensing Letters*, 12(2):329–333, 2015.
5. Xiaofei He, Deng Cai, and Partha Niyogi. Laplacian score for feature selection. In *NIPS*, 2005.
6. Ji-ming Li and Yun-tao Qian. Clustering-based hyperspectral band selection using sparse nonnegative matrix factorization. *Journal of Zhejiang University - Science C*, 12:542–549, 07 2011.

7. W. Sun, L. Zhang, B. Du, W. Li, and Y. Mark Lai. Band selection using improved sparse subspace clustering for hyperspectral imagery classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6):2784–2797, 2015.
8. M. Zeng, Y. Cai, Z. Cai, X. Liu, P. Hu, and J. Ku. Unsupervised hyperspectral image band selection based on deep subspace clustering. *IEEE Geoscience and Remote Sensing Letters*, 16(12):1889–1893, 2019.
9. Pan Ji, Tong Zhang, Hongdong Li, Mathieu Salzmann, and Ian Reid. Deep subspace clustering networks. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
10. M. Zeng, Y. Cai, X. Liu, Z. Cai, and X. Li. Spectral-spatial clustering of hyperspectral image based on laplacian regularized deep subspace clustering. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, 2019.
11. Xiao-Jiao Mao, Chunhua Shen, and Yu-Bin Yang. Image denoising using very deep fully convolutional encoder-decoder networks with symmetric skip connections. *CoRR*, abs/1603.09056, 2016.
12. Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, pages 585–591. MIT Press, 2002.
13. D. Cai, X. He, J. Han, and T. S. Huang. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1548–1560, 2011.
14. X. Ma, W. Liu, S. Li, D. Tao, and Y. Zhou. Hypergraph p -laplacian regularization for remotely sensed image recognition. *IEEE Transactions on Geoscience and Remote Sensing*, 57(3):1585–1595, 2019.
15. Y. Dai, B. Xu, S. Yan, and J. Xu. Study of cardiac arrhythmia classification based on convolutional neural network. *Computer Science and Information Systems*, 17(2):445–458, 2020.
16. Andrew Ng, Michael Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. *Adv. Neural Inf. Process. Syst*, 14, 04 2002.
17. Jianbo Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
18. E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2765–2781, 2013.
19. J. F. C. Mota, J. M. F. Xavier, P. M. Q. Aguiar, and M. Püschel. D-admm: A communication-efficient distributed algorithm for separable optimization. *IEEE Transactions on Signal Processing*, 61(10):2718–2723, 2013.
20. Lu Canyi, Jiashi Feng, Zhouchen Lin, Tao Mei, and Shuicheng Yan. Subspace clustering by block diagonal representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP:1–1, 01 2018.
21. Qi Liu, Miltiadis Allamanis, Marc Brockschmidt, and Alexander Gaunt. Constrained graph variational autoencoders for molecule design. 05 2018.
22. J. A. Dabin, A. M. Haimovich, J. Mauger, and A. Dong. Blind source separation with l1 regularized sparse autoencoder. In *2020 29th Wireless and Optical Communications Conference (WOCC)*, pages 1–5, 2020.
23. G. Spigler. Denoising autoencoders for overgeneralization in neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):998–1004, 2020.
24. Pan Ji, M. Salzmann, and Hongdong Li. Efficient dense subspace clustering. In *IEEE Winter Conference on Applications of Computer Vision*, pages 461–468, 2014.
25. Xiao Jiao Mao, Chunhua Shen, and Yu Bin Yang. Image denoising using very deep fully convolutional encoder-decoder networks with symmetric skip connections. *CoRR*, abs/1603.09056, 2016.

26. F. Luo, B. Du, L. Zhang, L. Zhang, and D. Tao. Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image. *IEEE Transactions on Cybernetics*, 49(7):2406–2419, 2019.
27. G. Cheng and X. Tong. Fuzzy clustering multiple kernel support vector machine. In *2018 International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR)*, pages 7–12, 2018.
28. S. Rogic and L. Kascelan. Class balancing in customer segments classification using support vector machine rule extraction and ensemble learning. *Computer Science and Information Systems*, 18(3):893–925, 2021.

Meng Zeng received the B.Sc. from the Hubei University of Arts and Science, Xiangyang, China, in 2017 and received the M.S. degree in computer science with the China University of Geosciences, Wuhan, China, in 2020. She is currently pursuing the Ph. D. degree in School of Computer Science and Engineering, Central South University, Changsha, China. Her current research interests include machine learning, hyperspectral image processing, data mining and storage.

Bin Ning received the Master degree in software engineering from Beijing University of Technology, Beijing, China, in 2005. He is currently a Professor with the School of Computer Engineering, Hubei University of Arts and Science, Xiangyang, China. His research interests include software engineering, data mining, and algorithm design. He is a member of the China Computer Federation.

Qiong Gu received the M.S. degree in Computer science and technology from China University of Geosciences, in 2006,and the Ph.D. degree in Geosciences Information Engineerin from China University of Geosciences, Wuhan, China, in 2009. She is a member of the China Computer Federation. She is currently a professor with the School of Computer engineering, Hubei University of Arts and Science, She is particularly interested in Data mining, Machine Learning, she has extended her interests include Net-Mediated Public Sentiment, and internet of things.

Chunyang Hu received the Ph.D. degree in computer science from Beihang University, Beijing, China, in 2011. He is currently an associate professor with the school of computer engineering, Hubei University of Arts and Science, Xiang yang, China. He served as a visiting scholar at University of Massachusetts Lowell in 2017. His research interests include big data processing, machine learning, software defined network, and block chain. Dr. Hu is a member of China Computer Federation.

Shuijia Li received the B.Sc. from the Hubei University of Arts and Science, Xiangyang, China, in 2017. He is currently pursuing the Ph. D. degree in computer science with the China University of Geosciences, Wuhan, China. His current research interests include evolutionary computation, computational intelligence and its application.

Received: August 30, 2021; Accepted: March 05, 2022.

Computer Science and Information Systems 19(2):803–827 https://doi.org/10.2298/CSIS210920006D

Optimized Placement of Symmetrical Service Function Chain in Network Function Virtualization

Nhat-Minh Dang-Quang[1] and Myungsik Yoo[2]

[1] Department of Information Communication Convergence Technology, Soongsil University
Seoul 06978, Korea
minhdqn@soongsil.ac.kr
[2] School of Electronic Engineering, Soongsil University
Seoul 06978, Korea
myoo@ssu.ac.kr

Abstract. Network function virtualization (NFV) is one of the key technology enablers for actualizing 5G networks. With NFV, virtual network functions (VNFs) are linked together as a service function chain (SFC), which provides network functionality for the customer on demand. However, how to efficiently find a suitable placement for VNFs regarding the given objectives is an extremely difficult issue. The existing approaches assume that the SFC has a simple and asymmetrical pattern that is unsuitable to modeling a real system. We address this limitation by studying a VNF placement optimization problem with symmetrical SFCs that can support both symmetric and asymmetric traffic flows. This NP-hard problem is formulated as a mixed-integer linear programming (MILP) model. An iterative greedy-based heuristic is proposed to overcome the complexity of the MILP model. Extensive simulation results show that the proposed heuristic can obtain a near-optimal solution compared to MILP for a small-scale network, and at the same time, is superior to a traditional heuristic for a large-scale network.

Keywords: Network function virtualization, multi-objective, VNF placement optimization, symmetric, heuristic.

1. Introduction

Network function virtualization (NFV) [1] has emerged as a new network paradigm that can overcome the limitations of traditional networks. NFV is a key emerging technology in multi-access edge computing (MEC) realizing the Internet-of-Things (IoT) and 5G networks [2]. Virtual customer premises equipment (vCPE) [3], which is the most popular use case of NFV, can provide a flexible platform where multiple network services (e.g., a firewall, router, dynamic host configuration protocol (DHCP), network address translation (NAT), and load balancing) are virtualized as virtual network functions (VNFs) and run on a common hardware platform. Through vCPE, service providers can rapidly develop new services and avoid all manual processes.

A network service usually consists of various VNFs based on the customer requirements. An ordered sequence of VNFs is formed through a service function chain (SFC) [4,5]. Some challenges having a significant impact on NFV include the efficient deployment of an SFC while ensuring that the service level agreements are satisfied and wisely allocating network resources. This is known as the VNF placement (VNF-P) problem,

which is the focus of this study. As mentioned earlier, an ordered sequence of VNFs is formed as a service function chain (SFC). Given a set of requested SFCs, the goal of VNF-P is to place the VNFs on suitable physical nodes in the network with regard to the given objectives while satisfying constraints related to the nodes, edge capacities, and latency bound [8,9,13,14,15,19,20]. A good placement solution may considerably enhance network resource usage efficiency and lower CAPital EXpenditures and OPerating EXpenses (CAPEX/OPEX), resulting in increasing profitability for cloud service providers. Given different service requests by different users, the VNF placement challenge concerns how to deploy a sequence of SFCs, each of which contains numerous VNFs, into cloud available resources. Several open-source NFV platforms, including OpenStack Tacker[3], Open Source MANO (OSM)[4], Open Platform for NFV (OPNFV) [5], and Open Network Automation Platform (ONAP)[6], have also concentrated on this problem.

In practice, SFC may be asymmetric or symmetric, depending on the requirement of the service providers. A symmetrical SFC can process a given two-way flow, i.e., a request flow (e.g., from a client to a server of the network service) and a response flow (e.g., from a server of the network service to a client)[4]. The existing studies on traffic symmetry for SFC are limited [6]. A hybrid SFC has attributes of both symmetric and asymmetric SFCs; that is, some VNFs require symmetric traffic, whereas other VNFs do not require response traffic or are independent of the corresponding request traffic [5]. However, conventional approaches assume that network services have a relatively simple and asymmetric paradigm, which is unsuitable to the modeling of real systems. Common VNFs, such as a deep packet inspection (DPI) or firewall are required to process symmetric traffic flows because of the consistent state of the flow [4,7]. In addition, to reduce wasted resources and minimize the number of deployed VNFs, the VNF instances can be reused across several SFCs in the network [19,20].

Based on the above observations, we propose a VNF-P model for SFCs that can support both symmetric and asymmetric traffic flows. This model is formulated as a mixed-integer linear programming (MILP) model with an objective function that simultaneously minimizes the number of deployed VNF instances, the total required data rates, and the total latency. In the model, the VNF instances can also be shared across several SFCs to reduce the number of deployed VNF instances. A heuristic based on the iterative greedy algorithm is presented to solve the problem in large-scale networks owing to the complexity of MILP. The simulation results show that the proposed heuristic can obtain a near-optimal solution compared to the MILP for a small-scale network and is also superior to its counterpart for a large-scale network.

The remainder of this paper is organized as follows. Section 2 overviews previous related studies. Section 3 describes the problem formulation and its model. Section 4 presents an iterative greedy-based heuristic method for solving the problem. Section 5 shows the simulation settings and numerical results. Finally, section 6 provides some concluding remarks.

[3] https://docs.openstack.org/tacker
[4] https://osm.etsi.org/
[5] https://www.opnfv.org/
[6] https://www.onap.org/

2. Related work

Since the concept of NFV was introduced in 2012 [11], the VNF-P began to draw attention as the building block of SFCs. SFC placement usually consists of two-step process: first, the resource allocation problem; second, the traffic steering problem. The VNF-P problems have been widely studied in recent years. In the majority of survey works, the VNF-P problems have been formulated as an integer programming (ILP, MIP or MILP) model. Then, heuristic placement algorithms have been proposed [12]. Furthermore, we see that the goal of SFC placement is the objective function of the optimization problem. Here are frequently used goals:

- Quality of Service (QoS) parameters: QoS parameters include energy consumption, service latency, availability, etc. These parameters can help the service provider to know the quality of a network service which provided to users.
- Cost and Revenue: The network cost represents the deploying cost or operating cost of VNF on the nodes. Meanwhile, the revenue is the net value earned by serving traffic needs, optimized under capacity constraints.

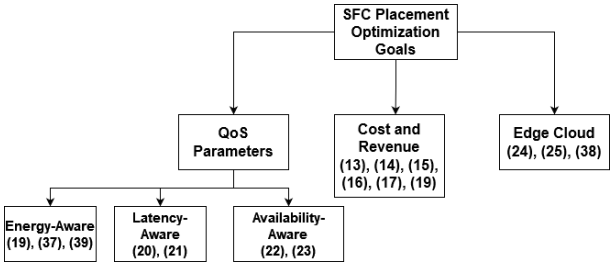


Fig. 1. Related works to VNF-P

Figure 1 shows three common optimization goals used for VNF placement optimization problem such as QoS parameters, cost and revenue and edge cloud. In this section, we categorize the related studies based on these goals.

- For the QoS parameters: if the deployment VNFs are not done properly in a compute resource-sharing environment like cloud computing, QoS will have a substantial impact on overall cloud service performance. A QoS-aware VNF placement strategy would significantly minimize traffic transmission across the whole data center, and therefore congestion and data transfer time. There are three common parameters in QoS-aware:
  - Energy-aware: It aims to minimize the power consumptions, which is achieved through policies from Service level agreements (SLAs).
  - Latency-aware: It aims to minimize network latency, VNF migration delay, etc.
  - Availability-aware: It aims to maximize the availability of VNFs.
- For the Cost and Revenue: This is the basic and fundamental resource allocation problem in NFV [10]. The main purpose of this problem is to minimize the total network cost, traffic volume in network.

– For the edge cloud: The concept of Mobile Edge Computing (MEC) brings the computing resource closer to end-users. Applying NFV to MEC will help reducing the service time latency for end-users as well as helping service providers to get more benefit from lower expenditures and higher efficiency [10].

Firstly, QoS can significantly impact the overall performance of cloud services if the VNF-P is not studied well. We categorized it into three common sub-category optimization goals, including Energy-Aware, Latency-Aware, and Availability-Aware.

For the energy-ware, in [19], the authors proposed a Monte Carlo Tree Search (MCTS) based method that shares VNFs among the tenants to minimize the energy consumption of the servers in the VNF-P model. Abdelaal et al. [37] proposed a novel approach for VNF placement called VNFRP (Virtual network functions and their replica placement). They formulated the VNF placement problem as an integer linear programming problem to optimize energy consumption and SFC placement cost. Furthermore, they proposed a heuristic-based algorithm to solve the proposed problem. The simulation showed when the number of replicas is increased, VNFRP may dramatically enhance load balancing by up to 80%. Zeng et al. [39] proposed a VNF placement and routing technique to optimize network delay and energy consumption. The technique combines a classic genetic algorithm with a simplex approach with strong local search capabilities, avoiding the problem that traditional genetic algorithms are prone to falling into the local optimum solution. The results showed that the suggested technique is capable of reducing network latency efficiency while also reducing network energy usage.

For the latency-aware, the authors [21] formulated the VNF placement problem as an ILP model and proposed a hidden Markov Chain-based heuristic for placing the VNFs to optimize the cost and delay. Agarwal et al. [22] formulated the VNF placement and CPU allocation as a convex optimization problem. Then, they proposed a method, which was based on the MaxZ placement heuristic, to optimize VNF placement and CPU allocation decisions.

For the availability-aware, Zhao and Dán [23] formulated the VNF placement as an ILP. Then they split it into a master problem and a sub-problem. They assumed that there are U failure scenarios, and each failure scenario i has a probability p_i. The master problem and sub-problem have been solved iteratively by using Generalized Benders Decomposition (GBD). In each iteration, they produced an upper bound and lower bound for the objective value of the original problem. The iteration process stops when these bound values meet the termination condition. The study [24] solved the end-to-end delay and service chaining availability for VNF placement using an ILP and heuristic solution.

Secondly, we surveyed the literature about cost-aware resource allocation in VNF-P. This is the basic and fundamental resource allocation problem in NFV [10]. The authors [13] found the number of essential VNFs and allocate them to minimize the total network cost and the resource fragmentation. The authors [13] presented an integer linear programming (ILP) model and a dynamic programming-based heuristic for the VNF-P problem. In [14], the VNF-P model was proposed using mixed-integer linear programming (MILP), which considers standard and fast path VNF forwarding methods with two different optimization goals, including traffic engineering and NFVI cost minimization. In [15], the target of the VNF-P model was to minimize the overall traffic volume in the network, whereas some VNFs can change the traversing traffic volume, e.g., a WAN optimizer can compress the traffic before sending it to the next hop, resulting in a change

in traffic volume. Pham et al. [16] proposed an algorithm based on the sampling Markov approximation for optimizing the operation and network traffic cost. Tomasasilli et al. [17] proposed two logarithmic factor approximation algorithms to optimize the deployment cost. The first algorithm was based on LP rounding, while the second algorithm was based on greedy algorithm. The authors in [20] analyzed the resource consumption on the servers and links. Their model allows different SFCs to share a single VNF if these SFCs demand the same VNF.

Thirdly, we also reviewed some works related to VNF-P in Edge cloud environment. Cziva et al. [25] formulated the VNF placement problem to minimize the total latency of all users to their VNFs. The authors [7] also applied the Optimal Stopping Theory to detect when to re-evaluate the optimal problem using two parameters: migration cost and path delay. Song et al. [26] put the study on the VNF placement for 5G edge computing using users' mobility. First, [26] proposed a user grouping model based on geographic information of user context and then defined (and calculated the optimal number of) clusters to minimize the delay of network services from one end to the other. Next, a graph partitioning algorithm that assigns VNFs to clusters was presented to minimize the movement between the user and clusters while optimizing the loss of users' data rate due to VNF migration. Tao et al. [38] formulated VNF placement problem to optimize both latency and energy consumption at the edge. A cost-minimizing optimization strategy is used to ensure the latency and energy consumption parameters. To overcome this challenge, they created a graph of edge systems and users. The placement of the VNFs is determined by the edge systems based on cost and user requirements. The findings demonstrate that the proposed VPE technique minimizes the cost of edge systems while maintaining the quality of the mobile user experience.

For a broader scope of resource allocation in NFV and its details, refer to previous comprehensive surveys [2,8,9,10].

The existing studies assumed that the SFC is asymmetric, which only considers a unidirectional traffic flow. Some Service Functions (SFs) need bidirectional flow, which means they required both forward and backward directions. For example, common VNFs such as deep packet inspection (DPI) or firewall are required to process symmetric traffic flows because of the consistent state of the flow. Thus, this is the limitation of the existing studies on VNF placement optimization problem due to it is not suitable to model real system. Thus, the existing studies on traffic symmetry are still limited.

There are few existing works that study symmetry SFC. Bifulco et al. [18] works on scalability and traffic steering for legacy mobile networks. Their proposal mentioned symmetry SFC, where the upstream and downstream traffics are the forward traffic and the backward traffic, respectively. The symmetry was achieved by using Network Address Translation (NAT). Their study took into account the overall symmetry of the chain, regardless of whether it is symmetric or asymmetric. Hantouti and Benamar [6] discuses benefits of using partially SFCs and introduce a new method for calculating the reverse route of symmetric SFCs. The result of the proposed method showed that it could help lower the state of forwarding and, as a result, the traffic delay. Therefore, the studies [6] and [18] do not focus on VNF placement for symmetric SFC. Thus, they are not presented in Figure 1.

By addressing mentioned limitations, this work studies the VNF placement optimization problem that can support both symmetric (unidirectional) and asymmetric (bidirec-

tional) traffic flows, which is suitable for modeling real systems. Our contribution is to formulate symmetric-enabled VNF placement optimization problem in order to minimize the number of deployed VNF instances, data required rate of SFCs, and total latency. Considering symmetry traffic for SFCs also helps save the number of deployed VNF instances because the VNF instances can be shared across several service functions. Thus, it is necessary to study on the VNF placement optimization for symmetrical SFC.

3. Symmetric-enabled VNF Placement Problem

In this section, the system model and problem formulation are presented. Table 1 shows the notations used in this paper.

3.1. System Model

NFV-enabled Network A directed graph is used to describe an NFV-enabled network, namely, $G = (N, E)$, where N denotes the sets of nodes and E denotes the sets of edges. We represent the CPU and memory capacities of each network node $n \in N$ as n_{cpu} and n_{mem}, respectively. A data rate capacity associated with each network edge $e \in E$ is represented as e_{dr}. Each network edge e has a latency e_{lat}.

Service function chain (SFC) A service function chain is specified by $S = (I_S, P_S)$. Here, $i \in I_S$ denotes a VNF instance of an SFC. A VNF instance can be implemented in a virtual machine (VM) or container running over the network infrastructure. In such a case, a specific amount of resources, such as the CPU and memory, are required. Hence, i_{cpu}, i_{mem} represent the CPU and memory resource consumption for a VNF instance i, respectively. The CPU and memory consumption for the VNFs can have a uniform distribution or other specific distribution depending on the service providers. The directed path between two instances defined in an SFC is denoted by $p \in P_S$, which connects exactly a head instance p_{head} of p to a tail instance p_{tail} of p. In addition, p_{Mlat} denotes the maximum latency that can be tolerated for path p.

In this study, an SFC can process both asymmetric and symmetric traffic flows, which require different types of VNF instances.

Type of VNF instances The four types of VNF instances considered are as follows.

(i) A source instance i_{src} represents a client. Therefore, it is fixed at a specific location and does not consume any CPU or memory resources. The source instance is given an outgoing data rate f_S^{dr} of a flow $f_S \in F_n$, where F_n is a union of all flows of SFCs at node n and $F_n \subset F$, where F is a union of all flows.

(ii) A symmetrical instance i_{sym} can process the symmetric traffic flows, i.e., the request (rq.) flow from a client to a server and response (rsp.) flow from a server to a client. The existing studies on the VNF-P problem assume that the SFCs only have one type of VNF instance, which processes only the asymmetric traffic flows. However, a VNF instance may be symmetrical or asymmetrical depending on the network service requirements. Many common VNFs such as a deep packet inspection (DPI), firewall, and L4-L7 load balancer often require a symmetric traffic flow processing feature to ensure that the flow state is consistent[4,7].

Table 1. Notations Used in the Paper

Notation	Description
Parameters	
$G = (N, E)$	A directed graph including a set of nodes N and a set of edges E.
$n_{cpu}, n_{mem} > 0$	CPU and memory capacities of node $n \in N$, respectively.
$e_{lat}, e_{dr} > 0$	Latency and data rate capacity of edge $e \in E$, respectively.
$S = (I_S, P_S)$	SFC including a set of instances $i \in I_S$ and a set of paths $p \in P_S$.
p_{head}, p_{tail}	Head and tail instances of path p.
$p_{Mlat} > 0$	Maximum latency of p that can be tolerated.
$i_{cpu}, i_{mem} > 0$	CPU and memory requirements of an instance.
i_{src}, i_{sym}, i_{asym}, i_{dst}	Role of instances, including source, symmetrical, asymmetrical, and destination instances.
r_i^{rq}, r_i^{rsp}	Request and response data rate scaling of an instance i.
F_n	A collection of all flows corresponding to all SFCs at node n.
F	A union of all flows, i.e., $F_n \subset F$.
$f_S \in F_n$	A flow from a source instance of an SFC S at node n. Each f_S has a data rate value of f_S^{dr}.
$0 \leq w_1, w_2, w_3 \leq 1, w_1+w_2+w_3 = 1$	Weighting factors of objectives.
Auxiliary variables	
$ingress_{i,n}^{f_S,rq}$, $ingress_{i,n}^{f_S,rsp} \geq 0$	Incoming data rate of request and response flows f_S of instance i placed at node n.
$egress_{i,n}^{f_S,rq}$, $egress_{i,n}^{f_S,rsp} \geq 0$	Outgoing data rate of request and response flows f_S of instance i placed at node n.
r_i^{rq}	data rate scaling of instance i for request flow
r_i^{rsp}	data rate scaling of instance i for response flow
Decision variables	
$y_{i,n} \in \{0, 1\}$	1 if an instance i is placed at node n.
$z_{n_1,n_2}^{p,e} \in \{0, 1\}$	1 if an edge e is used for sending traffic of path p that has p_{head} placed at n_1 and p_{tail} placed at n_2.
$dr_{n_1,n_2}^{p,e} \geq 0$	Required data rate at an edge e using the path p for sending traffic when the path p has p_{head} placed at n_1 and p_{tail} placed at n_2.

(iii) An asymmetrical instance i_{asym} only processes either a request or a response flow. Therefore, it has only incoming/outgoing data rates of either a request or response flow.

(iv) A destination instance i_{dst} redirects the traffic flow from an incoming request to an outgoing response. Therefore, it has the incoming data rates of the request flow and outgoing data rates of the response flow. In this paper, the destination can be the server of a network service or content caching server, which can be flexibly deployed in the network.

Each VNF instance has the incoming request/response flows and the outgoing request/response flows based on the type. We define four data rate values of a VNF instance i placed at node n of a flow f_S, i.e., incoming request data rate $ingress_{i,n}^{f_S,rq}$, outgoing request data rate $egress_{i,n}^{f_S,rq}$, incoming response data rate $ingress_{i,n}^{f_S,rsp}$, and outgoing response data rate $egress_{i,n}^{f_S,rsp}$. The required data rate at an edge of a flow may be changed when the flow passes a VNF instance [15]. For example, the WAN optimizer VNF compresses the traffic before sending it to the next hop, resulting in a traffic savings of up to 80% [27], the video optimizer VNF can decrease the data rate by up to 50% owing to a video trans-rating procedure [28], or the content filtering VNF can reduce the required data rate by blocking the video streaming during working hours [29]. Therefore, we define the data rate scaling of instance i for the request and response flows, i.e., $r_i^{rq} = \frac{egress_{i,n}^{f_S,rq}}{ingress_{i,n}^{f_S,rq}}$ and $r_i^{rsp} = \frac{egress_{i,n}^{f_S,rsp}}{ingress_{i,n}^{f_S,rsp}}$, respectively. Figure 2 shows four types of VNF instances with their flows and data rates. Figure 3 illustrates a sample of a symmetric-enabled SFC.

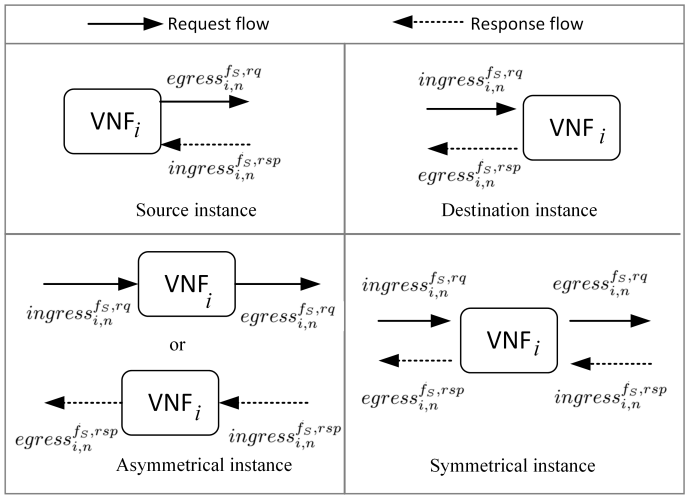


Fig. 2. Four types of VNF instances with their flows and data rates

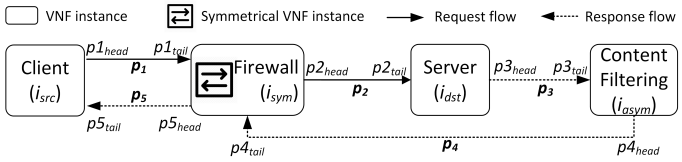


Fig. 3. A sample of symmetric-enabled SFC

Shared VNF The model also adapts the concept of the sharing and reuse of VNFs [19,20], which allows the different SFCs to use the same VNF instances with the same identifier in their flows. The VNF reuse strategy can improve the resource utilization of the servers and reduce the number of VNF instances deployed. A VNF instance can have the request and/or response flows for every SFC using that instance. This means there are sets of incoming and outgoing data rates corresponding to each SFC at a shared instance. All instances except for the source instance can be shared.

Figure 4 shows an example of symmetrical SFCs placed in the Abilene network and sharing an instance in which the source instance of SFC 1 is fixed at node 7, and the source instance of SFC 2 is fixed at node 2. The sample SFC consists of a source instance (Client - CLT), a firewall instance (FW), a server instance (SVR), and a content filtering instance (CF).

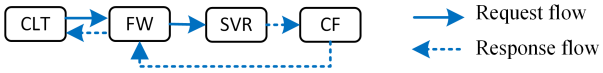
3.2. Problem Formulation

The objective of the VNF-P problem is to find the optimal location of VNF instances of symmetric-enabled SFCs in the network such that the number of VNF instances required, the data rate of the SFCs required, and the total latency caused by the SFCs are simultaneously minimized while satisfying the constraints related to the node and edge capacities, as well as the latency bounds for each SFC.

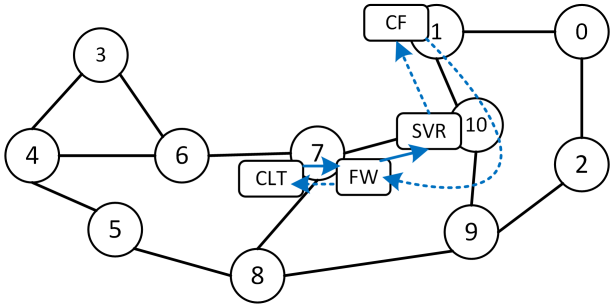
Variable Declaration The VNF-P problem is formulated as a mixed-integer linear programming (MILP) [30] model. The decision variables of the model are detailed as follows.

(i) The binary variable $y_{i,n}$ represents a placement of the instance i at node n.
(ii) The binary variable $z^{p,e}_{n_1,n_2}$ represents whether an edge e is used for sending traffic of path p, which has a head instance p_{head} placed at node n_1 and a tail instance p_{tail} placed at node n_2.
(iii) The continuous variable $dr^{p,e}_{n_1,n_2}$ represents a required data rate of path p at an edge e if that edge is used for sending traffic of path p, which has a head instance p_{head} placed at node n_1 and a tail instance p_{tail} placed at node n_2.

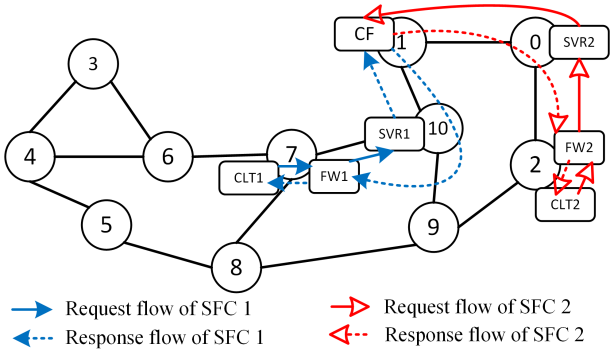
In addition, the continuous variables $ingress^{f_S,rq}_{i,n}$, $ingress^{f_S,rsp}_{i,n}$, $egress^{f_S,rq}_{i,n}$, and $egress^{f_S,rsp}_{i,n}$, which are auxiliary variables, are data rate values of an incoming/outgoing request/response corresponding to flows f_S of an instance i placed at node n, respectively.



A sample of symmetric-enabled SFC.



A sample SFC placed in the network.



Two sample SFCs sharing an instance.

Fig. 4. An example of symmetric-enabled SFCs placed in the network

Objective Function and Constraints The optimization objective is to simultaneously minimize the number of VNF instances placed across the network (obj_1), the total required data rate (obj_2), and the total latency (obj_3). The problem can be formulated following the MILP model.

$$\begin{aligned}&\text{minimize}\\&w_1 \cdot obj_1(\mathbf{x}) + w_2 \cdot obj_2(\mathbf{x}) + w_3 \cdot obj_3(\mathbf{x})\\&\text{subject to } (1) - (15)\end{aligned}$$

where

(i) $\mathbf{x} = (y_{i,n}, dr^{p,e}_{n_1,n_2}, z^{p,e}_{n_1,n_2})$, which is a decision vector in a feasible set. The feasible set is defined by the following constraints.
(ii) $obj_1(\mathbf{x}) = \sum_{i \in I_S, n \in N} y_{i,n}$. This function defines the total number of VNF instances deployed in the network.
(iii) $obj_2(\mathbf{x}) = \sum_{p \in P_S, n_1, n_2 \in N, e \in E} dr^{p,e}_{n_1,n_2}$. This function defines the total data rate required by all SFCs in the network.
(iv) $obj_3(\mathbf{x}) = \sum_{p \in P_S, n_1, n_2 \in N, e \in E} (z^{p,e}_{n_1,n_2} \cdot e_{lat})$. This function defines the total latency of all SFCs in the network.
(v) w_1, w_2, and w_3 are the weighting factors in which $0 \leq w_1, w_2, w_3 \leq 1$, and $w_1 + w_2 + w_3 = 1$, and are used to assign the importance of the functions based on the specific requirement of the service providers.

The following constraints are considered in the model. Constraints (1) and (2) are the mapping consistency rules for the source instance. In (1), every source instance is placed in a predefined node. The outgoing data rate of every source instance equals the predetermined data rate of the flow exiting from that instance, as shown in (2).

$$\forall S, \forall i \in I_S, \text{if } i \text{ is } i_{src}, \exists! n \in N : y_{i,n} = 1 \tag{1}$$

$$\begin{aligned}&\forall S, \forall i \in I_S, \forall n \in N, \text{if } y_{i,n} = 1, \text{if } i \text{ is } i_{src},\\&\exists! f_S \in F_n : egress^{f_S,rq}_{i,n} = f^{dr}_S\end{aligned} \tag{2}$$

Constraints (3), (4), and (5) express the data rate rules of an instance i when placed at node n. In (3), if an instance is i_{sym} or i_{asym}, for a request flow, the outgoing data rate of that instance equals the r^{rq}_i scaling rate of the incoming data of that instance. In (4), if an instance is i_{sym} or i_{asym}, for a response flow, the outgoing data rate of that instance equals the r^{rsp}_i scaling rate of the incoming data of that instance. In (5), if an instance is i_{dst}, the outgoing data rate of the response flow of that instance equals the r^{rq}_i scaling rate of the incoming data of the request flow of that instance.

$$\begin{aligned}&\forall S, \forall i \in I_S, \forall n \in N, \forall f_S \in F, \text{if } y_{i,n} = 1,\\&\text{if } i \text{ is } i_{sym} \text{ or } i_{asym} : egress^{f_S,rq}_{i,n} = r^{rq}_i \cdot ingress^{f_S,rq}_{i,n}\end{aligned} \tag{3}$$

$$\begin{aligned}&\forall S, \forall i \in I_S, \forall n \in N, \forall f_S \in F, \text{if } y_{i,n} = 1,\\&\text{if } i \text{ is } i_{sym} \text{ or } i_{asym} : egress^{f_S,rsp}_{i,n} = r^{rsp}_i \cdot ingress^{f_S,rsp}_{i,n}\end{aligned} \tag{4}$$

$$\begin{aligned}&\forall S, \forall i \in I_S, \forall n \in N, \forall f_S \in F, \text{if } y_{i,n} = 1,\\&\text{if } i \text{ is } i_{dst} : egress_{i,n}^{f_S,rsp} = r_i^{rq} \cdot ingress_{i,n}^{f_S,rq}\end{aligned} \tag{5}$$

Constraints (6) and (7) show the data rate rules of the head and tail instances of path p. The incoming data rate of tail instance p_{tail} of path p equals the outgoing data rate of a head instance p_{head} of that path correlated with the request and response flow f_S.

$$\begin{aligned}&\forall S, \forall p \in P_S, \forall n_1, n_2 \in N, \forall f_S \in F,\\&\text{if } y_{p_{head},n_1} = y_{p_{tail},n_2} = 1 : egress_{p_{head},n_1}^{f_S,rq}\\&= ingress_{p_{tail},n_2}^{f_S,rq}\end{aligned} \tag{6}$$

$$\begin{aligned}&\forall S, \forall p \in P_S, \forall n_1, n_2 \in N, \forall f_S \in F,\\&\text{if } y_{p_{head},n_1} = y_{p_{tail},n_2} = 1 : egress_{p_{head},n_1}^{f_S,rsp}\\&= ingress_{p_{tail},n_2}^{f_S,rsp}\end{aligned} \tag{7}$$

Constraint (8) expresses the flow conservation in which the flow must leave an egress of an instance if the flow passes through it. The required data rate at every edge along a path p equals the outgoing data rate of a head instance of that path over all flows.

$$\begin{aligned}&\forall S, \forall p \in P_S, \forall n, n_1, n_2 \in N, \text{if } y_{p_{head},n_1} = y_{p_{tail},n_2} = 1 :\\&\sum_{nn' \in E} dr_{n_1,n_2}^{p,nn'} - \sum_{n'n \in E} dr_{n_1,n_2}^{p,n'n} =\\&\begin{cases}\sum_{f_S} egress_{p_{head},n_1}^{f_S,rq} \text{ if } n = n_1 \neq n_2, f_S \text{ is request.}\\\sum_{f_S} egress_{p_{head},n_1}^{f_S,rsp} \text{ if } n = n_1 \neq n_2, f_S \text{ is response.}\\0 \text{ otherwise}\end{cases}\end{aligned} \tag{8}$$

Constraint (9) ensures the consistency of the variables $z_{n_1,n_2}^{p,e}$ and $dr_{n_1,n_2}^{p,e}$. If a path p uses an edge e for sending traffic, a required data rate of that path exists at that edge. Otherwise, it does not.

$$\begin{aligned}&\forall S, \forall p \in P_S, \forall n_1, n_2 \in N, \forall e \in E :\\&z_{n_1,n_2}^{p,e} = \begin{cases}1 & \text{if } dr_{n_1,n_2}^{p,e} > 0\\0 & \text{otherwise}\end{cases}\end{aligned} \tag{9}$$

Constraint (10) prevents a loop in the path. It states that a path p should only use one direction of an edge if that edge is used for sending traffic.

$$\begin{aligned}&\forall S, \forall p \in P_S, \forall n_1, n_2 \in N, \forall nn' \in E,\\&\text{if } n'n \in E : z_{n_1,n_2}^{p,n'n} + z_{n_1,n_2}^{p,nn'} \leq 1\end{aligned} \tag{10}$$

Constraint (11) guarantees that the total latency of the edges used by a path p cannot surpass the maximum latency of that path.

$$\forall S, \forall p \in P_S, \forall n_1, n_2 \in N : \sum_{e \in E} (z_{n_1,n_2}^{p,e} \cdot e_{lat}) \leq p_{Mlat} \tag{11}$$

Constraint (12) ensures that a request flow and the corresponding response flow must go through the same symmetrical instance i_{sym}.

$$\begin{aligned}
&\forall S, \forall i \in I_S, \forall n \in N, \forall f_S \in F, \text{if } i \text{ is } i_{sym}: \\
&\text{if } ingress_{i,n}^{f_S,rq} + egress_{i,n}^{f_S,rq} > 0: \\
&ingress_{i,n}^{f_S,rsp} + egress_{i,n}^{f_S,rsp} > 0 \\
&\text{if } ingress_{i,n}^{f_S,rq} + egress_{i,n}^{f_S,rq} = 0: \\
&ingress_{i,n}^{f_S,rsp} + egress_{i,n}^{f_S,rsp} = 0
\end{aligned} \tag{12}$$

Constraints (13), (14), and (15) are capacity constraints. In (13), the required data rates at an edge cannot exceed the data rate capacity of that edge. The node resource capacity constraints are shown in (14) and (15).

$$\forall S, \forall e \in E: \sum_{p \in P_S, n_1, n_2 \in N} dr_{n_1,n_2}^{p,e} \leq e_{dr} \tag{13}$$

$$\forall S, \forall n \in N: \sum_{i \in I_S} (i_{cpu} \cdot y_{i,n}) \leq n_{cpu} \tag{14}$$

$$\forall S, \forall n \in N: \sum_{i \in I_S} (i_{mem} \cdot y_{i,n}) \leq n_{mem} \tag{15}$$

Model Complexity With the help of optimization solvers (e.g., Gurobi [32], CPLEX [33]), the optimal solution can be achieved using a combinatorial search (e.g., Branch-and-Bound algorithm [34]). However, the VNF-P problem has been proven to be NP-hard [8,13]. This means that the time complexity increases exponentially with the network size. Therefore, it becomes a challenge to obtain an optimal solution when a network is large.

4. Proposed Heuristic-based Placement Method

Because the VNF placement problem is an NP-hard problem, which means that there is no algorithm that will always efficiently produce the exactly correct answer on all inputs. To address the NP-hard complexity of the problem, we propose a heuristic to solve it. The benefits of the proposed Heuristic-based Placement method is to find a feasible (not optimal) solution that is good enough to quickly solve and achieve optimization placement goals. The heuristic includes three steps. First, the edges in the network are weighted based on the latency and data rate capacity. Second, an initial solution is given by a greedy algorithm based on the calculated weight of the edges. Finally, an iterative greedy algorithm improves the initial solution using a random placement process.

4.1. Heuristic-based placement process

Every edge e is assigned a weight $e_w = \frac{1}{e_{dr}} + e_{lat}$ and all pairs with the shortest (least-weight) paths in the network are found using the Floyd-Warshall algorithm [31] concurrently in which an edge with a higher data rate capacity and lower latency has a lower weight compared to the other edges.

A greedy algorithm aims to select a node for placing a new VNF instance if that node has sufficient capacity available and has the least-weight path from the current node of the VNF instances. The greedy algorithm is illustrated in Algorithm 1. The instances are processed following the sequence of the flow in both directions in the SFC. The source instance is placed to a predefined node. Each instance i has a path p that connects the prior instance to it. The algorithm finds prospective nodes for deploying the instance i corresponding to p. A node is a prospective node if it satisfies the node, edge capacity, and latency constraints. To support shared instances, a node can be a prospective node if it has deployed instances that are the same type as the considered instance i, and satisfies the edge capacity and latency constraints. If there is no prospective node, the algorithm returns an infeasible solution. Otherwise, the algorithm creates or reuses (if exists) the instance i at a prospective node that has the least-weight path from the location of the prior instance. It should be noted that if the request flow passes an instance i_{sym}, the corresponding response flow must return exactly to that instance.

Algorithm 1 Greedy algorithm pseudocode

Input: Weighted $G = (N, E)$; $\forall S$; $i_{rand} = null$.
Output: Placement Solution Sol.

for all S **do**
  Place i_{src} at a predefined location.
  for all other i in I_S in both directions of flow f_S **do**
    Get path p coming to instance i in flow f_S.
    if f_S is in response direction & i is i_{sym} **then**
      Map p to least-weight path connected node of i_{sym}.
    else
      Find prospective nodes that satisfy node, edge capacities, and latency constraints.
      if perspectives nodes do not exist **then**
        Return infeasible solution.
      end if
      Select a prospective node that has the least-weight path from the location of the prior instance.
      Create or reuse (if exists) instance i on the selected node and map p to the corresponding least-weight path.
      Update node and edges capacities.
    end if
  end for
end for
return Solution Sol

The solution of the greedy algorithm may be local optimal because the greedy algorithm simply chooses the minimum-weight path with the corresponding end node to place a new VNF instance. An iterative greedy algorithm is proposed to avoid the local optimality of the greedy algorithm, as shown in Algorithm 2. Given the initial solution by the greedy algorithm and predefined maximum number of iterations, the iterative greedy algorithm arbitrarily chooses an instance in the current solution and places it to a different location. For every iteration, the algorithm creates a new solution, which has a new

objective value calculated using the objective function. The algorithm then compares two objective values of the two solutions, chooses the solution with the least objective value, and assigns it as the best solution. This process is repeated with the current best solution until the maximum number of iterations is reached. The algorithm will return the solution with the lowest objective value.

Algorithm 2 Iterative greedy algorithm pseudocode

Input: Weighted $G = (N, E)$; $\forall S$; Sol; n_iter.
Output: Best solution Sol_{best}

```
1: Sol_best ← Sol
2: iter ← 0
3: while iter < n_iter do
4:     iter ← iter + 1
5:     Select a random instance i_rand from Sol_best except the source instances.
6:     Sol_new ← run greedy algorithm but assign i_rand at a different node.
7:     obj_best ← calculate the objective value of Sol_best
8:     obj_new ← calculate the objective value of Sol_new
9:     if obj_new < obj_best then
10:        Sol_best ← Sol_new
11:    end if
12: end while
13: return Sol_best
```

4.2. Complexity Analysis

The time complexity of the placement process is a combination of the phases, including finding all pairs of shortest paths using the Floyd-Warshall algorithm and repeating the greedy algorithm in the number of iterations. The Floyd–Warshall algorithm takes $O(N^3)$ [31]. An efficient implementation of the greedy algorithm is used to compute the placement of an instance, taking $O(N \log N)$. For an SFC, it takes $O(IN \log N)$, where I is the maximum number of instances needed to be considered in both flow directions. Therefore, for K number of SFCs, the greedy algorithm takes $O(KIN \log N)$. By contrast, the iterative greedy algorithm repeats the greedy algorithm in the number of iterations M. Therefore, it takes $O(MKIN \log N)$. Hence, the overall running time of the heuristic-based placement process is $O(N^3 + MKIN \log N)$.

5. Performance Evaluation

In this section, extensive simulations conducted to verify the proposed model and algorithms are described.

5.1. Simulation Settings

The algorithms under evaluation are the proposed MILP (denoted as **MILP**), the proposed heuristic algorithm using 20 iterations (as described in section Conclusion, and denoted

as **proposed heuristic**), and the first-fit heuristic algorithm (a baseline algorithm, denoted as **firstfit heuristic**), which takes each instance in turn and places it into the first node that can accommodate it. Gurobi optimizer 8.1 [32] using a Branch-and-Bound algorithm [34] is used to solve the MILP model. Python 3 programming language was used to develop the simulation and algorithms. All computations were conducted on a PC supplied with an Intel Xeon CPU E3-1230 V2 @ 3.30 GHz, 16 GB RAM, running Windows 10 x64 OS.

The SFC used for the evaluation is a content-filtering service, as illustrated in Fig. 5. It consists of a (Client - CLT), a firewall instance (FW), a server instance (SVR), and a content filtering instance (CF). First, CLT sends requests to FW. The FW is responsible for analyzing inbound and outbound network traffic and chooses whether to allow or prohibit certain types of traffic based on a set of predefined rules. It also supports symmetry traffic flows, which means it always receives and responds to requests from CLT. If the request is valid, the FW forwards the request to SVR to process. After processing the request of CLT, the SVR sends the response to CF. The CF blocks content that contains harmful information, such as pornographic content, etc. Finally, it sends back the response to FW to return to CLT. We assume that the CPU and memory consumption of each VNF instance obey a uniform distribution of (2,4). The data rate scaling equals 1 for all instances except for the CF instance, which has a data rate scaling r_i^{rsp} of 0.5. The maximum latency that can be tolerated for each path p_{Mlat} is 20 ms. Every SFC has a required outgoing data rate of $f_S^{dr} = 1$ (Gbps) from its source instance for all simulations.

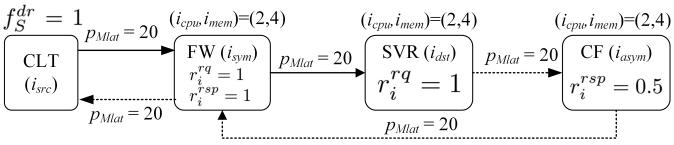


Fig. 5. Symmetric-enabled content filtering SFC used for the evaluation

The simulations were conducted on the Abilene network, which is an ISP backbone network, and the Geant network, which is a research backbone network. Both networks were taken from the Internet topology zoo dataset [35]. In these networks, the edge latency e_{lat} in milliseconds is estimated as the propagation latency calculated from the geographical distances between nodes [36]. Table 2 shows the details of the networks used and their setups.

The inputs of the experiment are the network, SFC and number of SFCs need to be deployed. We evaluate the ouput of proposed heuristic-based placement method with other methods by four objectives: Number of deployed VNFs instances, Total required data rate measured in giga bytes per second (Gbps), Total latency measured in second (s) and Computational time measured in second (s).

Table 3 shows the four settings of the weighting factors. Using the balance setting of the weighting factor, we first look at the effect of the symmetric feature of the SFCs in the proposed model as compared to a conventional model. Then, the optimal results achieved using the MILP of the proposed model with the four different settings of the weighting factors are examined. The number of SFCs increases from 1 to 5 owing to the

Table 2. Network topologies used for simulations

Network	$\|N\|$	$\|E\|$	e_{dr} (Gbps)	n_{cpu} (Unit)	n_{mem} (GB)
`Abilene`	11	28	10	4	8
`Geant`	40	122	20	4	8

Table 3. Weighting factor settings

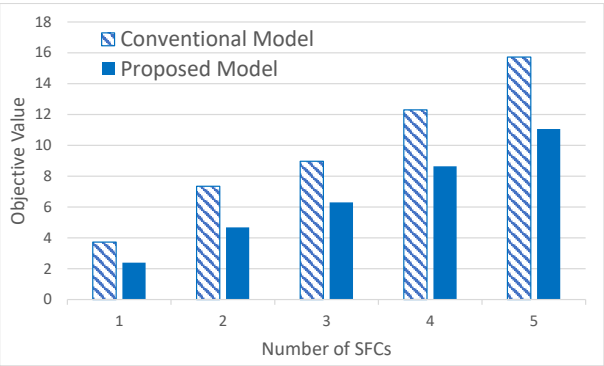
Scenarios	w_1	w_2	w_3
`MILP-Balance`	1/3	1/3	1/3
`MILP-VNF`	1	0	0
`MILP-Dr`	0	1	0

exponential computational time of the MILP. Using a balanced setting of the weighting factors, the MILP is then compared with the proposed heuristic and the first-fit heuristic. The proposed heuristic is then compared with the first-fit heuristic in the large-scale Geant network with the number of SFCs varying from 1 to 20. Every SFC has a different source instance location from the others.

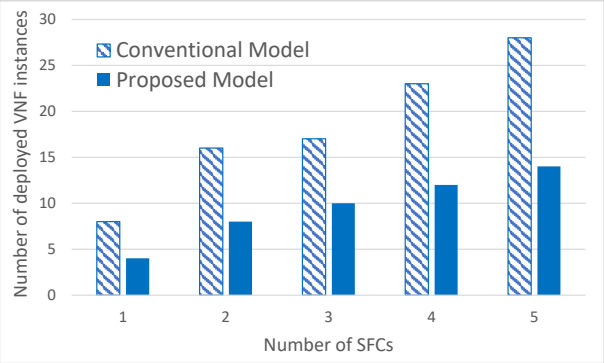
5.2. Simulation Results

Conventional model versus proposed model One disadvantage of the conventional model, which does not support the symmetric feature of the VNF instances, is that it cannot ensure a consistent flow state. By contrast, if the model does not support symmetrical VNF instances, the given SFC should be separated into sub-SFCs, i.e., (i) $CLT \rightarrow FW \rightarrow SVR$, (ii) $SVR \rightarrow CF \rightarrow FW$, and (iii) $FW \rightarrow CLT$. The additional instances need to be deployed to process these SFCs instead of the one-time process of the proposed model. This can lead to an increase in the number of deployed VNF instances required, which increases the node's resource consumption and server energy consumption. The simulation was conducted on the Abilene network. The results of the conventional and proposed models differ significantly only in the number of VNF instances deployed. The latency and data rate results are the same because the two models achieve the same optimal placement of the VNF instances. As shown in Fig. 6, the gap between the objective value and the number of VNF instances required between the two models increases with the number of SFCs.

Effect of the weighting factors Figure 7 illustrates the effect of different weighting factors on the MILP performance. MILP-VNF maintains the lowest number of deployed VNF instances by sharing all deployed VNF instances except for the source instances. However, it must sacrifice the data rate and latency objectives because the flow must go further to reach the shared instances. By contrast, MILP-Dr and MILP-Lat deploy numerous instances within the node capacities to reduce the data rate and latency, respectively. MILP-Balance tries to minimize all three objectives simultaneously because these three objectives have equal weighting factors. The results show that MILP-Balance can archive the optimal required data rate and total latency of MILP-Dr and MILP-Lat, where the number of deployed VNF instances is not overly high compared to MILP-VNF.

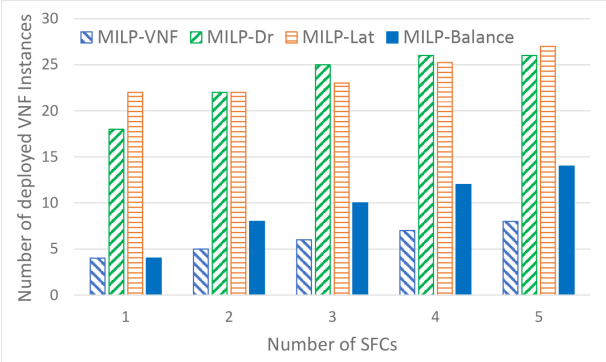


Objective value

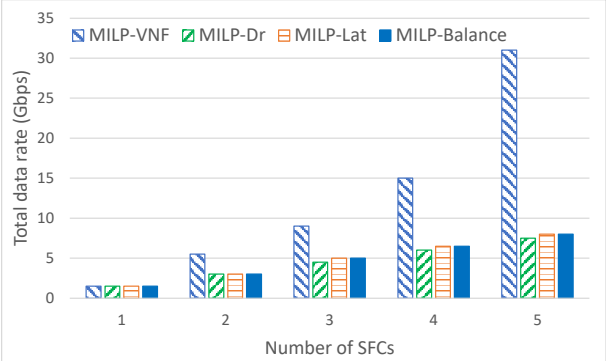


Number of deployed VNF instances

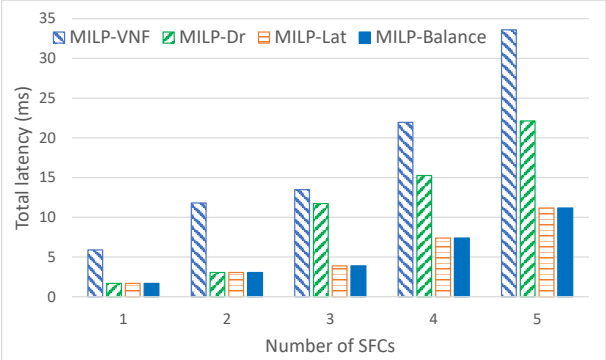
Fig. 6. Comparison of conventional and proposed models



Number of deployed VNFs instances



Total required data rate (Gbps)



Total latency (ms)

Fig. 7. The effect of different weighting factors on MILP performance in the Abilene network

Results of MILP and heuristics in the Abilene network From Fig. 8, it can be seen that the proposed heuristic outperforms the first-fit heuristic and achieves a near-optimal objective value compared to MILP with an average optimality gap of 9.7%. The details of each objective are shown in Fig. 9. The first-fit heuristic has the lowest number of deployed VNF instances among the different algorithms because it chooses the first node that can accommodate the VNF instance and reuse that instance in other SFCs. However, the first-fit heuristic must sacrifice the latency and data rates required because the distance between VNF instances is greater than that in the solution to the other algorithms. It also shows that the proposed heuristic can simultaneously minimize all objectives and effectively avoid being trapped in the local optimality compared to the first-fit heuristic. In terms of the computational time, as a simple algorithm, the first-fit heuristic outperforms the other algorithms. However, the proposed heuristic with 20 repetitions only takes approximately 1 s to solve the problem with five SFCs, whereas the MILP takes 10.5 h.

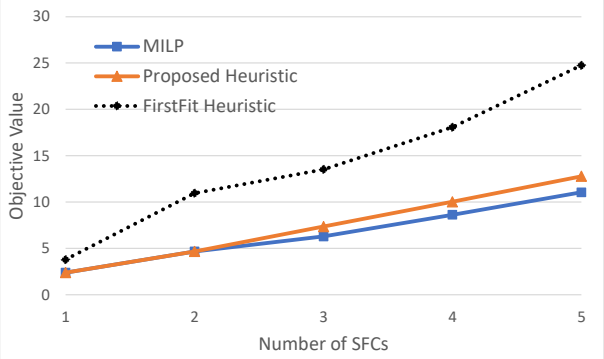
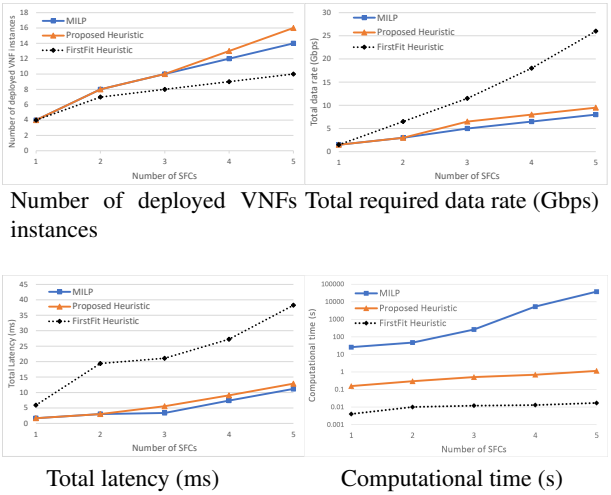


Fig. 8. Objective value of the algorithms in Abilene network

Results of heuristics in the Geant network With the increase in the size of the network and number of SFCs, the MILP cannot obtain the solution within an acceptable time. Therefore, we only compare the performance of the proposed heuristic and first-fit heuristic in the Geant network. The objective value given by the proposed heuristic is again always lower than the objective value given by the first-fit heuristic, as shown in Fig. 10. The gap between two objective values of the algorithms under evaluation increases when increasing the number of SFCs. Figure 11 shows the details of all objectives. The first-fit heuristic uses the same shared VNF instances and therefore keeps the number of VNF instances required as low as possible. However, the latency and data rate required by the first-fit heuristic are too high compared to that of the proposed heuristic. It should be noted that the first-fit heuristic cannot produce a solution because it violates the data rate capacity constraint when the number of SFCs is 20. The proposed heuristic undoubtedly has a higher computational time compared to the first-fit heuristic. However, it only takes under a minute (i.e., 42 s) when placing 20 SFCs at one time. However, with the quality of the solution determined by the proposed heuristic, this can be acceptable when



Number of deployed VNFs instances | Total required data rate (Gbps)

Total latency (ms) | Computational time (s)

Fig. 9. Performance comparison in Abilene network

running on a large network. The computational time of this heuristic can be reduced if we choose a suitable number of iterations and implement it using high-performance servers.

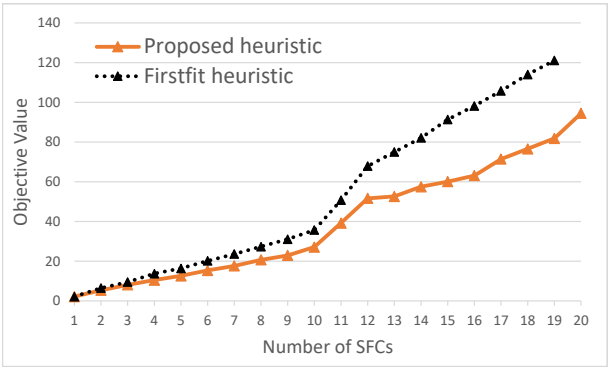
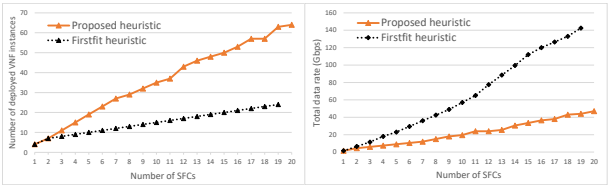


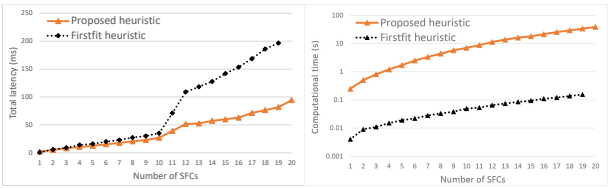
Fig. 10. Objective value of the iterative greedy algorithm and the greedy algorithm in Geant network

6. Conclusion

Conventional approaches assume that network services have a relatively simple and asymmetric paradigm, which is unsuitable to the modeling of real systems. Common VNFs, such as a deep packet inspection (DPI) or firewall are required to process symmetric traffic flows because of the consistent state of the flow [4,7]. The impact of this study is that



Number of deployed VNFs instances | Total required data rate (Gbps)



Total latency (ms) | Computational time (s)

Fig. 11. Performance comparison in Geant network

it focused on the VNF placement problem with reusable VNF instances for symmetric-enabled SFC, which can process both asymmetric and symmetric traffic flows. The symmetric features of SFC can not only ensure the consistency of the flow state but also reduce the number of deployed instances. This NP-hard problem was formulated using a multi-objective MILP model, which minimizes number of VNF instances, data rate of SFCs, and total latency. Owing to the complexity of the MILP model, an iterative greedy-based heuristic was proposed to solve the problem in large-scale networks. The benefits of the proposed Heuristic-based Placement method is to find a feasible (not optimal) solution that is good enough to quickly solve and achieve optimization placement goals. The extensive simulation results showed that the proposed heuristic can gain the near-optimal solution (under a 10% optimality gap) within a shorter time period compared to the MILP approach for a small-sized network. The performance of the proposed heuristic was also superior to the baseline first-fit heuristic for a large-sized network.

For future works, we plan to formulate the optimization problem for more real-world scenarios. We will conduct more experiments on other common networks used in VNF placement optimization problem such as AAR, JGN2plus, etc., which can be found in [35]. Also, due to the development of artificial intelligence, especially with Deep Reinforcement Learning (DRL) in solving VNF-P problem, the authors will apply this technique to solve the optimization problems.

Acknowledgments. This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support programs (IITP-2021-0-02046) and (IITP-2021-2017-0-01633) supervised by the IITP (Institute of Information & Communications Technology Planning & Evaluation)

References

1. ETSI NFV, "Network function virtualisation: An introduction, benefits, enablers, challenges & call for action," Introductory White Paper, Issue 1, *SDN and OpenFlow World Congress*, Darmstadt, Germany, Oct. 2012.
2. B. Yi, W. Xingwei, L. Keqin, and H. Min, "A comprehensive survey of network function virtualization," *Comput. Netw.*, Vol. 133, pp. 212-262, 2018.
3. T.H. Nguyen, T. Nguyen and M. Yoo, "Analysis of deployment approaches for virtual customer premises equipment," in *Proc. 32th IEEE Int. Conf. Inform. Netw. (ICOIN 2018)*, pp. 289-291, 2018.
4. P. Quinn and T. Nadeau, "Problem statement for service function chaining," No. RFC 7498. 2015.
5. J. Halpern and C. Pignataro, "Service function chaining (sfc) architecture," No. RFC 7665. 2015.
6. H. Hantouti and N. Benamar, "Partially Symmetric Service Function Chains," 2019 2nd IEEE Middle East and North Africa COMMunications Conference (MENACOMM), 2019, pp. 1-6, doi: 10.1109/MENACOMM46666.2019.8988534.
7. C. Zhang, S. Addepalli, N. Murthy, L. Fourie, M. Zarny, and L. Dunbar, "L4-L7 Service Function Chaining Solution Architecture," Open Networking Foundation, ONF TS-027, 2015.
8. J.G. Herrera and J.F. Botero, "Resource allocation in NFV: A comprehensive survey." *IEEE Trans. Netw. Service Manag.*, Vol. 13, No. 3, pp. 518-532, 2016.
9. A. Laghrissi, and T. Tarik, "A Survey on the Placement of Virtual Resources and Virtual Network Functions," *IEEE Commun. Surveys Tuts.*, 2018.
10. S. Yang, F. Li, S. Trajanovski, R. Yahyapour and X. Fu, "Recent Advances of Resource Allocation in Network Function Virtualization," in IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 2, pp. 295-314, 1 Feb. 2021, doi: 10.1109/TPDS.2020.3017001.
11. Network functions virtualisation: An introduction benefits enablers challenges & call for action, October 2012.
12. A. Mohamad and H. S. Hassanein, "On Demonstrating the Gain of SFC Placement with VNF Sharing at the Edge," 2019 IEEE Global Communications Conference (GLOBECOM), 2019, pp. 1-6, doi: 10.1109/GLOBECOM38437.2019.9014106.
13. F. Bari, R.C. Shihabur, A. Reaz, B. Raouf, and C.M.B.D Otto, "Orchestrating virtualized network functions," *IEEE Trans. Netw. Service Manag.*, Vol. 13, No. 4, pp. 725-739, 2016.
14. M. Gao, B. Addis, M. Bouet and S. Secci, "Optimal orchestration of virtual network functions," *Comput. Netw.*, Vol. 142, pp. 108-127, 2018.
15. W. Ma, S. Oscar, B. Jonathan, P. Deng, and P. Niki, "Traffic aware placement of interdependent nfv middleboxes." in *Proc. IEEE Conf. Comput. Commun. (INFOCOM 2017)*, pp. 1-9, 2017.
16. C. Pham, N. H. Tran, S. Ren, W. Saad, and C. S. Hong, "Trafficaware and energy-efficient vNF placement for service chaining: Joint sampling and matching approach," IEEE Trans. Services Comput., vol. 13, no. 1, pp. 172–185, Jan./Feb. 2020.
17. A. Tomassillik, F. Giroire, N. Huin, and S. Perennes, "Provably efficient algorithms for placement of service function chains with ordering constraints," in Proc. IEEE INFOCOM, 2018, pp. 774–782.
18. R. Bifulco, A. Matsiuk and A. Silvestro, "CATENAE: A scalable service function chaining system for legacy mobile networks", Int. J. Netw. Manag., vol. 27, no. 2, pp. 1-14, 2017
19. Soualah, Oussama, Marouen Mechtri, Chaima Ghribi, and Djamal Zeghlache. "Energy efficient algorithm for VNF placement and chaining." in *Proc. 17th IEEE/ACM Int. Symp. Cluster, Cloud and Grid Computing (CCGRID 2017)*, pp. 579-588, 2017.
20. T.W. Kuo, B.H. Liou, K.C.J Lin, and M.J. Tsai. "Deploying chains of virtual network functions: On the relation between link and server usage." *IEEE/ACM Trans. Netw.*, Vol. 26, No. 4, pp. 1562-1576, 2018.

21. H. Chen et al., “MOSC: A method to assign the outsourcing of service function chain across multiple clouds,” Comput. Netw., vol. 133, pp. 166–182, 2018.
22. S. Agarwal, F. Malandrino, C. Chiasserini and S. De, ”Joint VNF Placement and CPU Allocation in 5G,” IEEE INFOCOM 2018 - IEEE Conference on Computer Communications, 2018, pp. 1943-1951, doi: 10.1109/INFOCOM.2018.8485943.
23. P. Zhao and G. Dán, ”Resilient placement of virtual process control functions in mobile edge clouds,” 2017 IFIP Networking Conference (IFIP Networking) and Workshops, 2017, pp. 1-9, doi: 10.23919/IFIPNetworking.2017.8264849.
24. P. Vizarreta, M. Condoluci, C. M. Machuca, T. Mahmoodi and W. Kellerer, ”QoS-driven function placement reducing expenditures in NFV deployments,” 2017 IEEE International Conference on Communications (ICC), 2017, pp. 1-7, doi: 10.1109/ICC.2017.7996513.
25. R. Cziva, C. Anagnostopoulos and D. P. Pezaros, ”Dynamic, Latency-Optimal vNF Placement at the Network Edge,” IEEE INFOCOM 2018 - IEEE Conference on Computer Communications, 2018, pp. 693-701, doi: 10.1109/INFOCOM.2018.8486021.
26. S. Song, C. Lee, H. Cho, G. Lim and J. Chung, ”Clustered Virtualized Network Functions Resource Allocation based on Context-Aware Grouping in 5G Edge Networks,” in IEEE Transactions on Mobile Computing, vol. 19, no. 5, pp. 1072-1083, 1 May 2020, doi: 10.1109/TMC.2019.2907593.
27. Citrix, “Improve the XenApp and XenDesktop experience for branch and mobile workers with NetScaler SD-WAN.” [Online]. Available: https://www.citrix.com/content/dam/citrix/en_us/documents/products-solutions/improve-the-xendesktop-experience-for-branch-and-mobile-workers-with-netscaler-sdwan.pdf
28. Tellabs, “Mobile Video Optimization Concept and Benefits,” White Paper, 2011. [Online]. Available: https://s3.amazonaws.com/zanran_storage/www.tellabs.com/ContentPages/2438991029.pdf
29. WebtTitan, “Internet Content Filtering Service.” [Online]. Available: https://www.webtitan.com/internet-content-filtering-service/
30. C.A. Floudas, “Nonlinear and mixed-integer optimization: fundamentals and applications,” Oxford University Press, 1995.
31. T.H. Cormen, C.E. Leiserson and R.L. Rivest, “The floyd-warshall algorithm,” Introduction to Algorithms, 558, p.565, 1990.
32. Gurobi Optimization. [Online]. Available: http://www.gurobi.com/
33. IBM ILOG CPLEX Optimization Studio. [Online]. Available: https://www.ibm.com/analytics/cplex-optimizer
34. J. Clausen, “Branch and bound algorithms-principles and examples,” Department of Computer Science, University of Copenhagen, p.1-30, 1999.
35. S. Knight, H.X. Nguyen, N. Falkner, R. Bowden and M. Roughan, “The internet topology zoo.” *IEEE J. Sel. Areas Commun.*, Vol. 29, No. 9, pp.1765-1775, 2011.
36. J. Kurose, and R. Keith, “Computer networks and the internet.” Computer networking: A Top-down approach. 7th ed. London: Pearson, 2016.
37. Abdelaal, Marwa A. and Ebrahim, Gamal A. and Anis, Wagdy R., ”Efficient Placement of Service Function Chains in Cloud Computing Environments” Electronics, 2021, doi: 10.3390/electronics10030323.
38. Tao, Xiaoyi and Ota, Kaoru and Dong, Mianxiong and Qi, Heng and Li, Keqiu, ”Cost as Performance: VNF Placement at the Edge” IEEE Networking Letters, 2021, doi: 10.1109/LNET.2021.3065651.
39. Zeng, Ying and Shi, Zhan and Wu, Zanhong, ”VNF Placement and Routing Algorithm for Energy Saving and QoS Guarantee” Proceedings of the 9th International Conference on Computer Engineering and Networks, 2021, doi: 10.1007/978-981-15-3753-089.

Nhat-Minh Dang-Quang received a Master's degree in Information and Communication Technology (ICT) from Soongsil University, Seoul, South Korea in February, 2022. He also received a B.Eng. degree in Software Engineering from the University of Information Technology, Vietnam National University—Ho Chi Minh City, Ho Chi Minh City, Vietnam, in 2019. His research interests include cloud computing, auto-scaling, and self-healing.

Myungsik Yoo received B.S. and M.S. degrees in electrical engineering from Korea University, Seoul, South Korea, in 1989 and 1991, respectively, and a Ph.D. degree in electrical engineering from The State University of New York at Buffalo, New York, in 2000. He was a Senior Research Engineer with the Nokia Research Center, Burlington, MA. He is currently a Full Professor with the School of Electronic Engineering, Soongsil University, Seoul. His research interests include visible light communications, cloud computing, Internet protocols.

Received: September 20, 2021; Accepted: February 24, 2022.

Computer Science and Information Systems 19(2):829–856 https://doi.org/10.2298/CSIS210930017S

MEC-MS: A Novel Optimized Coverage Algorithm with Mobile Edge Computing of Migration Strategy in WSNs

Zeyu Sun[1,2,3], Guisheng Liao[1,3,4⋆], Cao Zeng[1,3,4], Zhiguo Lv[2], and Chen Xu[5]

[1] National Key Laboratory of Radar Signal Processing, Xidian University, Xian 710071, China.
lylgszy@163.com gsliao@xidian.edu.cn czeng@mail.xidian.edu.cn
{lzg96wl,xuchensit}@163.com
[2] School of Computer and Information Engineering, Luoyang Institute of Science and Technology, Luoyang 471023, China
[3] Collaborative Innovation Center of Information Sensing, Xidian University, Xian 710071, China
[4] International Coopreation Base of Integrated Electronic Information System of Ministry of Science and Technology, Xidian University, Xian 710071, China
[5] School of Computer Science and Information Engineering, Shanghai Institute of Technology, Shanghai 200235, China

Abstract. The traditional network coverage mode with the cost of deploying a large number of sensor nodes has poor coverage effect. Aiming at this problem, this paper proposes a Novel Optimized Coverage Algorithm with Mobile Edge Computing of Migration Strategy (MEC-MS). First, the algorithm uses the network coverage model to give the expression method of the distance measurement and the judgment conditions of the best and worst paths. Secondly, it analyzes the necessary conditions for improving the coverage quality and the prerequisite for the existence of redundant coverage for adjacent the redundant coverage nodes by the theory of probability. Thirdly, using the precondition of redundant coverage, we give the calculation process of the sensor nodes own redundant coverage and the calculation method of the redundant node coverage expectation. Finally, the algorithm compares the number of working sensor nodes with the other two algorithms under different parameters. The experimental results show that the average number of working sensor nodes in the MEC-MS algorithm is 9.74% lower than that of the other two algorithms, and the average value of network coverage is 9.92% higher than that of the other two algorithms, which verify the effectiveness of the algorithm in this paper.

Keywords: wireless sensor networks, mobile edge computing, migration strategy, optimization coverage,networks lifetime.

1. Introduction

Wireless sensor network is a large self-organizing network that organically integrates the physical and information world, and is also an important carrier for human society to perceive the physical world and the information world [1-5]. Wireless sensor networks have been widely used as the underlying perception system of the Internet of Things, and their applications are mainly concentrated in the fields of military and national defense, safety production, intelligent industry, environmental monitoring, emergency rescue and

⋆ Corresponding author

medical and health. Optimizing coverage is a key issue in the research field of wireless sensor networks. The quality of coverage directly affects the quality of other performance indicators of wireless sensor networks, such as random deployment, route backup, and target tracking [6-9]. In the process of optimizing coverage, the solution to the coverage problem is mainly focused on the following fields.

Firstly, how to use the least number of sensor nodes to cover the largest monitoring area under a random deployment environment. Secondly, how to effectively improve the coverage under the premise of meeting a certain connectivity coverage effective coverage of the target nodes of interest. Thirdly, how to resist the energy consumption of sensor nodes under the premise of ensuring transmission quality to extend the network life time. Fourth, in the monitoring area, when a large number of working sensor nodes are covered at the same time, how to avoid the emergence of redundant nodes to improve the coverage of the target nodes of interest [10-13]. Therefore, as a basic problem in wireless sensor network research, the coverage problem has attracted the attention of many scholars at home and abroad.

The traditional coverage mode in the sensor network is mainly with the underlying coverage based on the perception layer. The sensor nodes are static, and the basic state information of the target node is obtained through the perception ability. The main shortcomings areas follows. (1) The lack of control nodes in the network makes it impossible to effectively control and manage other sensor nodes. (2) Due to the lack of unified coordination capabilities, there will often be a large number of redundant nodes, thus forming a communication link congestion. (3) There is a phenomenon that a large number of nodes cover the target node at the same time, which speeds up the energy consumption of the node and reduces the accuracy of the effective coverage rate. (4) In the process of data transmission, due to the number of hops from the sensor node to the base station, The network responds slowly to data transmission, which leads to longer data delays and frequent data packet loss. Compared with traditional coverage methods, the advantages of the proposed MS-MEC algorithm in this paper are mainly reflected in the following. Firstly, other nodes can be effectively controlled, organized and managed through controllable nodes, thereby reducing redundant nodes, improving work efficiency and enhances the robustness of the network. Secondly, the state of sensor nodes is effectively scheduled through the node scheduling algorithm. Under the condition of no need to work, shut down some sensor nodes, suppress node energy consumption, and extend the network life time [14-16]. Thirdly, under the premise of meeting a certain effective coverage rate, the sensor nodes in the sensing coverage area can be reasonably scheduled to focus on the target in the process of data transmission, the MS-MEC algorithm in this paper can effectively reduce the number of data transmission link hops, suppress the generation of delay, reduce data packet loss, and improve data transmission effectiveness [17-19]. In general, the coverage problem is based on the coverage model, and the coverage accuracy are improved through a certain calculation method to achieve the purpose of extending the network life time. In terms of the entire monitoring area, we do not need to effectively cover the entire monitoring area, but to effectively cover the target node of interest to improve the accuracy of the coverage rate. In order to better study the problem of optimized coverage, this article puts forward the Optimized Coverage Algorithm with Mobile Edge Computing of Migration Strategy in WSNs (MEC-MS) which is based on the research background of mobile edge

computing and node migration strategy. The main contributions of this article include the following aspects.

(1) Chapter Two, this article mainly analyzes and studies related work, and gives the advantages and disadvantages of different documents. Solutions and measures are proposed for the deficiencies of related documents.

(2) In Chapter Three, this article gives relevant hypotheses and basic definitions, and then, establishes and analyzes the coverage model, and gives the analysis process at the same time. And also introduces relevant parameters to optimize the coverage model.

(3) In Chapter four, the process of sensor node coverage is calculated and reasoned by the change of related parameters and the implementation and analysis process of the MS-MEC algorithm are also presented.

(4) In Chapter five, the characteristics of the sensor network is compared by simulation experiments, and the comparison experiment process and analysis process are given, so as to further illustrate the effectiveness and stability of the MS-MEC algorithm in this paper.

(5) Chapter six summarizes the full text and points out the key tasks in the future.

2. Related Works

As an important subject in the field of sensor network research, the coverage problem has attracted the attention of many scholars at home and abroad, and a series of related studies have been carried out, and some very important results have also been achieved. With the rapid development of theoretical knowledge and practical applications of sensor networks, higher requirements are put forward for sensor network coverage [20-21]. Such as, how to quickly and accurately cover the moving target node, how to achieve unified control of heterogeneous sensor networks and effective coverage of local locations in the monitoring area, etc. In the process of collecting data, sensor nodes are forced to make a large number of rapid responses to deal with various situations due to the diversity of data and some irresistible factors in the process of data transmission. Only by effectively covering the target node can we provide accurate data for our next research.

2.1. Linear Coverage Problem

Literature [22] uses the sensor nodes perception ability to directly cover the target node. When the target node moves at random time, the sensor node wakes up its neighbor nodes through perception control to complete the continuous coverage of the target node. Literature [23] proposed an Event-driven Mechanism Coverage Algorithm Based on Sensing-cloud-computing in Heterogeneous Sensor Networks (EMC-SC). Firstly, the algorithm uses the network coverage model to calculate the boundary relationship between the peer square and the sensor node coverage area. Secondly, the authors analyze the coverage performance of randomly deployed sensor nodes through the Poisson distribution model, and further calculate the probability of effective coverage of the monitored area. Thirdly, the coverage rate of the target node is determined according to the distance relationship between neighbor nodes and sensor nodes and the energy of the remaining sensor nodes. Literature [25] uses the neighboring nodes perception ability to linearly cover the target node, and uses a geometric-based calculation method to complete the monitoring area coverage, and at the same time, the solving process for the required minimum number of

sensor nodes is present. Literature [26] uses a virtual grid division method to effectively divide the monitoring area and uses time series to periodically cycle the coverage period of sensor nodes. The purpose of this algorithm is to find the best matching position information in each time sequence to achieve effective coverage of the target node. After N cycles, the algorithm uses an adaptive function to adjust the optimal distance between the sensor node and the target node to achieve accurate coverage, thereby it improves the adaptive capability and coverage efficiency of WSN. Literature [27] adjusts the sensing range by adjusting the transmit power of sensor nodes to achieve coverage. The algorithm first judges the distance relationship between the sensor node and the target node. If the distance between the sensor node and the target node is more than twice the sensing radius, the sensor nodes transmitting power is increased, and vice versa. When the transmit power is greater than or equal to the threshold, and the target node still cannot be covered, the sensor node is turned off to save energy. Literature [28] divides the monitoring area into multiple equilateral triangle areas while discussing effective coverage, so that any sensor node is at the apex of the equilateral triangle and gives the proof process of the relevant maximum coverage theorem. Literature [29] proposed a division method that divides the monitoring area into multiple concentric circles and squares. Through the parameter relationship between the sensing radius and the communication radius, the connected coverage ratio function is given and proved, and the maximum coverage rate of the monitoring area is finally obtained. Literature [30] proposed a diamond-shaped coverage model in which each sensor node is placed on the vertex of the diamond to ensure the monitoring of the diamond-shaped coverage area. The sensor nodes perception and communication capabilities can achieve effective coverage of the diamond-shaped coverage area. For the entire area, iterating the above process will complete the effective coverage of the entire area. Literature [31] discussed a maximum coverage algorithm based on the dominating set under the premise of satisfying the connected coverage rate. The algorithm completes the effective coverage of the monitoring area by means of adjacent nodes forming an adjacency graph, and combines the characteristics of random deployment to give the network topology of the mobile sink node in the data collection process. Literature [32] proposed a data-centric coverage control algorithm. The algorithm uses the idea of data center based on the multiple relationship of the step length of the perception radius, gradually expands to the periphery, and finally achieves effective coverage of the whole world through the collaborative work of adjacent nodes. Literature [33] studied the energy conversion mechanism under the premise of satisfying the coverage rate. The energy conversion process and algorithm realization process of sensor nodes in the coverage process are given. This model successfully completes the energy conversion of sensor nodes and prolongs the network life time.

2.2. Non-linear Coverage Problem

Literature [34] discussed the continuous coverage of target nodes along a certain trajectory. The proposed algorithm proves the constraint condition of the continuous coverage of sensor nodes, and to a certain extent ensures the local effective coverage of the entire monitoring area. Literature [35] gives a centralized k-degree coverage model. The model proves the solution process of the minimum number of sensor nodes required for k under different value ranges, and proves that the functional relationship between the communication radius and the sensing radius when the monitoring area is effectively covered,

and the expected value of the number of sensor nodes is formulated by using the sensor node density. Literature [36] takes the network life time as the research background and proposes a discrete coverage algorithm based on k degree. The algorithm sets the conditions for the existence of the upper and lower limits of the discrete coverage rate. In terms of network energy consumption, the algorithm uses the state transition mechanism of sensor nodes to schedule sensor nodes in different states to complete effective coverage of target nodes. Literature [37] completed the effective coverage of the target node with integer programming method. The algorithm is based on a heuristic algorithm and studies the collective membership of the sensor node set and the target node to realize the heuristic coverage algorithm. At the same time, the high-density sensor node is used to construct a coverage set to complete the coverage of the moving target node. Literature [38] uses artificial bee colony and particle swarm algorithm to convert different states of sensor nodes, thereby completing the uninterrupted conversion of sensor node states and realizing the process of covering moving targets. The algorithm also gives the best plan for deploying sensor nodes. Literature [39] proposes an efficient and energy-saving connectivity coverage routing algorithm. The algorithm divides the monitoring area twice, uses the knowledge of probability theory to calculate the network coverage, and finally achieves effective coverage of the moving target node. Literature [40] uses ant colony algorithm to optimize the deployment of sensor nodes. Through the ant colony algorithm, the entire network is continuously traversed and searched. And then the partial optimization of the entire monitoring area is realized, thereby expanding to the entire network monitoring area.

3. Related Definitions and Network Models

In order to better study the network coverage problem and simplify the complexity of calculation, this paper makes the following assumptions.

(1) All sensor nodes have a unique identification ID and show a disc shape.

(2) In the monitoring area, the sensor node density is large enough and boundary effects are ignored.

(3) The length of the boundary of the monitoring area is much longer than the sensor nodes sensing radius.

(4) The location information of all sensor nodes is acquired through a certain positioning algorithm, Such as: RSSI positioning algorithm.

Definition 1:(Sensing neighbor set) The set formed by the distance between any two sensor nodes less than or equal to the sum of the sensing radius.

$$S(n) = \{(i,j) \,|d(i,j) \leq r_i + r_j, i \neq j\} \tag{1}$$

where $S(n)$ is the set of perceptual neighbors, $d(i,j)$ is the distance between any two sensor nodes, r_i and r_j are the sensing radius of sensor node *i* and sensor node *j*, respectively.
Definition 2: (Joint Coverage Ratio) In the monitoring area, the ratio of the coverage area of all sensor nodes to the area of the monitoring area.

$$p\left(s,A\right)=\sum_{i=1}^{n}s_i\Big/S\left(A\right) \tag{2}$$

where s_i is the area covered by any sensor node, n is the number of sensor nodes; $S(A)$ represents the area of the monitoring area.

Definition 3: (Effective coverage area) In the monitoring area, the difference between the area of any sensor node and the coverage area of adjacent sensor nodes.

$$S_{ec}=\sum_{i=1,j=1}^{n}\left(\left(s_i\cup s_j\right)-\left(s_i\cap s_j\right)\right) \tag{3}$$

where s_i and s_j respectively represent the area covered by any sensor node i and sensor node *j*, *n* represents the number of sensor nodes, S_{ec} represents the effective coverage area.

Definition 4: (Effective Coverage Rate) The ratio of the effective coverage area to the area of the monitoring area.

$$p_{ec}=\sum_{i=1,j=1}^{n}\left(\left(s_i\cup s_j\right)-\left(s_i\cap s_j\right)\right)\Big/S\left(A\right) \tag{4}$$

Definition 5: (Network Life time) The longest time for the sensor network to work and run is called the network life time.

Definition 6: (Perception probability) According to the sensor nodes perception ability, the probability that any target node in the monitoring area is covered by the sensor node.

$$p=\begin{cases}0 & r_i\le d\left(s_i,s_j\right)\\ \varepsilon^{-\alpha d} & r_i-r_e<d\left(s_i,s_j\right)<r_i\\ 1 & d\left(s_i,s_j\right)\le r_i-r_e\end{cases} \tag{5}$$

where, $d(s_i,s_j)$ represents the Euclidean distance between the sensor nodes s_i and s_j, α represents the physical characteristic parameter, and r_e represents the variable parameter of the sensor node in the monitoring area.

3.1. Network Model Analysis

When a moving target node traverses the monitoring area, each sensor node on its traversing path will cover or monitor it. When the moving target node is traveling, the coverage density of each sensor node to the moving target node is different. The deployment of high-density sensor nodes indicates that the moving target node can be covered in the network with a higher probability, conversely, the low-density sensor node has a low coverage probability for moving target nodes, which further shows that moving target nodes are not easily exposed in a low-density deployment environment. Therefore, seeking an effective continuous coverage is an important subject of sensor network research, which is shown in Figure 1.

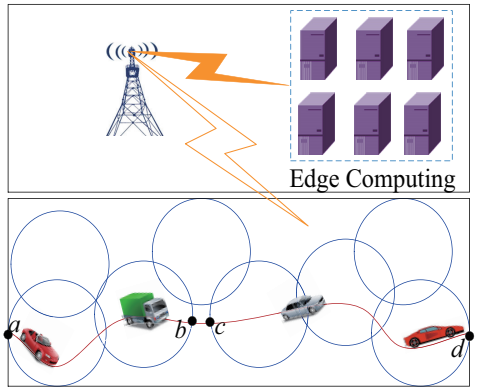


Fig. 1. The exposed discontinuous overlay network model

Figure 1 shows the exposed discontinuous overlay network model. It can be seen from Figure 1 that the trajectory of the moving target node is $a \rightarrow b \rightarrow c \rightarrow d$, and the moving target can be continuously covered by the sensor node in $a \rightarrow b$ and $c \rightarrow d$, and the moving target node in $b \rightarrow c$ is exposed to discontinuous coverage.

An effective way to solve the problem of sensor network coverage is to use the sensory characteristics of the sensor network to effectively cover the moving target node. Therefore, this paper proposes an optimized coverage algorithm (MEC-MS) based on mobile edge computing migration strategy. In the traditional coverage problem, the characteristic relationship between the coverage performance and the network life time is mainly studied. However, due to the limitation of the network space dimension, the traditional method cannot establish the complete geometric relationship between the multi-dimensional targets, and the research faces some uncertainty. In geometric space, the inner and outer products of vectors are expressed as follows.

$$a \cdot b = (ab + ba)/2 \tag{6}$$

$$a \times b = (ab - ba)/2 \tag{7}$$

where a and b are space vectors, $\times$ is not a multiplication sign, but also denoted as $\wedge$, the vertical component of the vector a is as follows.

$$a_{\perp} = a \times B/B \tag{8}$$

where $a_{\perp}$ is the vertical component of vector a, which is also the orthogonal complement of vector a, B is a reversible homogeneous order, corresponding to the number of adjacent sensor nodes.

Definition 7: (distance measurement) In geometric space, the distance between any two sensor nodes can be expressed as follows.

$$dist\left(s_i, s_j\right) = \|d_i - d_j\| \tag{9}$$

where d_i and d_j can be expressed as follows.

$$d_i = d\left(i, j\right) - s_{i\perp} \tag{10}$$

$$d_i = d\left(i, j\right) - s_{j\perp} \tag{11}$$

Substituting formula (8), formula (10) and formula (11) into formula (9), we can get the following formula (12).

$$dist\left(s_i, s_j\right) = \left(s_{i\perp} - s_{j\perp}\right) \times B/B \tag{12}$$

For a single node, we can change the above formula (12)

$$dist\left(s_i, s_j\right) = \left(s_{i\perp} - s_{j\perp}\right) \times B/B \tag{13}$$

By the above analysis, it can be seen that for the spatial coverage model, the distance between any point s_i and the target node s_t can be expressed by formula (13). In addition, when the value of B increases, the neighboring nodes of the sensor node s_i also increase, thus it forms an uninterrupted coverage chain to complete continuous coverage of the moving target node. In order to further increase the flexibility and diversity of the formula, we introduce the proportional dimension λ ($\lambda \in [0,1]$) for formula (13). We can adjust the distance measure of formula (13) by the controllability of λ.

$$D' = s_i \times B/B + \lambda B \tag{14}$$

$$D = \left(s_i - s_{i\perp}\right) \times B/B + \lambda B \tag{15}$$

Theorem 1: In the network space sensor node set *S*, there is at least one best path located in the edge set of the Voronoi graph.

Proof: In the Voronoi graph of the cyberspace sensor node set *S*, there is an edge set S_q in the Voronoi graph. According to the circle circumcenter theorem, there is a sensor node s_k that must fall on the vertical line of the edge set. It can be seen from formula (14) that the distance of this point is the smallest and the orthogonal complement between the vector in the orthogonal set corresponding to any points on the side S_q and s_k is also the smallest. In the same way, the other two sides of the triangle will repeat the above operation, and the minimum value of orthogonal set will also be obtained. Therefore, in the sensor node collective *S*, there must be an optimal path located in the edge set of the Voronoi graph. The proof is complete.

Theorem 2: In the network space sensor node set *S*, at least one path with the worst gap is located in the edge set of Delaunay Triangulation.
Proof: As shown in Figure 2, assuming the $P_m \in$ Delaunay Triangulation, the sensor nodes s_i and s_j are respectively at the two ends of the diameter of the sensor node s_c, and their coverage area is a solid line area, and the sensing radius distance of the area is measured as $(d_i - d_j)^d/d$, where d is the average of the unit vector measurement.
Discussion 1: In the common area covered by s_i and s_j, if there are no other sensor nodes except s_c, then there are sensor nodes s_i and s_j that do not include the coverage 'sensing area of other sensor nodes. According to the nature of Delaunay Triangulation, the sensor node s_i and s_j must be connected by an edge of the Delaunay Triangulation, which contradicts the assumption.
Discussion 2: If there are other sensor nodes s_p in addition to s_c in the common area covered by s_i and s_j, the edge set can be regarded as ip and jp, and ip and jp can be substituted for ij, then ij is not the edge formed by the nodes with largest distance measure, i.e., p is not the worst gap path, which contradicts the assumption.
The proof is complete.

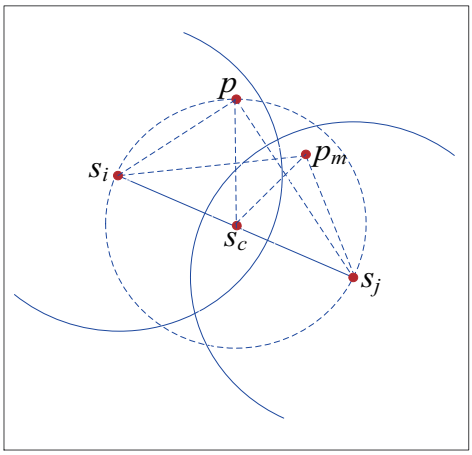


Fig. 2. Covering the worst gap path

4. Solution and Algorithm Analysis

Traditional sensor network coverage mainly achieves coverage of moving target nodes by adjusting the distribution of sensor node power to spatial location. Currently, network coverage is one of the basic problems that must be solved in network operation. Network interconnection mainly focuses on the connection between nodes, so that data can be transmitted from one node to another; while the coverage of sensor networks mainly reflects the sensor network's perception ability and network service quality. But its concerns are different with traditional network coverage. From the perspective of network

perception of the physical world, its main focus is on the location distribution of network nodes and the completion of effective coverage of target nodes under the conditions of coverage application. The main purposes of coverage control are as follows. (1) Optimizing sensor network coverage. (2) Reasonably allocating network space resources. (3) Better completing data perception and information collection. (4) Extend the network life time. With the help of the definition of perceptual probability, let us further analyze the proportional relationship between coverage membership function and coverage intensity.

4.1. Coverage Quality Analysis

Definition 8 (coverage intensity) In the monitoring area, the location information of a certain sensor node s_i is (x_i, y_i), and the location information of any target node p is (x, y), then the intensity from the coverage of the sensor node s_i to the target node p is defined as $I(s_i, p)$, which is represented by the membership function $F(x, y)$ as shown in formula (16).

$$I\left(s_i, p\right) = F\left(x, y\right) \tag{16}$$

Where the value range of $F(x, y)$ is in the closed interval of [0,1], and $F(x, y)$ reflects the coverage membership relationship of the target node p to the sensor node s_i.

Definition 9 (redundant coverage) In the monitoring area, the sensor set S composes with sensor nodes s_i, the proportion of the sensing area covering itself is called the redundant coverage.

$$p\left(s_i\right) = \frac{\sum_{s_i \in S}\left(s_i \cap s_j\right)}{s_i} \tag{17}$$

Since a large number of sensor nodes are randomly deployed in the monitoring area, and the Euclidean distance between any two or more adjacent sensor nodes is less than or equal to the sensing radius, and then the redundant nodes will appear. Then, after multiple cycles, the number of redundant nodes will greatly increase. When the redundant nodes with higher energy is started again, the redundant node will cover the target node only under certain limited conditions, i.e., when the redundant node changes from the sleep state to the working state, it must be greater than or equal to the current coverage threshold, otherwise the redundancy is still in the sleep state.

Theorem 3: The redundancy coverage of n adjacent redundant coverage nodes of sensor node s_i needs to satisfy the following condition.

$p\left(n\right) = 1 - \left\{1 - \frac{1}{\pi}\left[\frac{\pi}{4} - \frac{\lambda\sqrt{4-\lambda^2}}{8} + \frac{\lambda^2}{2}\arccos\left(\frac{\lambda}{2}\right) - \frac{\lambda^3\sqrt{4-\lambda^2}}{16}\right]\right\}^n$

Where λ is a controllable parameter, $\lambda \in [0, 1]$.

Proof. As shown in Figure 3, we suppose that the angle between any two sensor nodes B and C sensing radius lining and the intersection of the two circles is α. According to the uniform distribution characteristics of the redundant nodes, when the effective coverage area reaches the maximum value, the redundant coverage rate simultaneously reaches the maximum value. We suppose that the proportional relationship between the communication radius and the sensing radius is λ, i.e., $R = \lambda r$, λ is a controllable parameter. The distance L between the sensor node S_i and S_j is a random variable, which can be known

from the probability density formula as follows.

$$f_L(x) = \frac{2x}{\lambda^2 r^2} 0 \le x \le \lambda r \tag{18}$$

The area of the intersection area between the sensor node S_i and S_j is shown in formula (19).

$$S = (2\alpha - \sin 2\alpha) r^2 \tag{19}$$

The distance of the intersecting area is shown as follows.

$$l = 2r \cos\alpha \tag{20}$$

$$\mathrm{d}l = 2r \sin\alpha \mathrm{d}\alpha \tag{21}$$

where $\alpha \in \left[\arccos\frac{\lambda}{2}, \frac{\pi}{2}\right]$
We substitute formula (19) and (20) into the probability density formula can be obtained the following formula.

$$\begin{aligned} p_1 &= \int_{\frac{\lambda}{2}}^{\arccos\frac{\lambda}{2}} (2\alpha - \sin 2\alpha) \frac{4r^2 \cos\alpha}{\lambda^2} \sin\alpha \mathrm{d}\alpha \\ &= \frac{1}{\pi}\left[\frac{\pi}{4} - \frac{\lambda\sqrt{4-\lambda^2}}{8} + \frac{\lambda^2}{2}\arccos\frac{\lambda}{2} - \frac{\lambda^3\sqrt{4-\lambda^2}}{16}\right] r^2 \end{aligned} \tag{22}$$

The neighbor node redundancy coverage rate for any redundant node can be expressed as follows.

$$p_2 = \frac{p_1}{\pi r^2} = \frac{1}{\pi^2}\left[\frac{\pi}{4} - \frac{\lambda\sqrt{4-\lambda^2}}{8} + \frac{\lambda^2}{2}\arccos\frac{\lambda}{2} - \frac{\lambda^3\sqrt{4-\lambda^2}}{16}\right] \tag{23}$$

Under the knowledge of probability, when n sensor nodes cover any target node, the resulting redundant coverage is shown as follows.

$$p(n) = 1 - \left\{1 - \frac{1}{\pi^2}\left[\frac{\pi}{4} - \frac{\lambda\sqrt{4-\lambda^2}}{8} + \frac{\lambda^2}{2}\arccos\frac{\lambda}{2} - \frac{\lambda^3\sqrt{4-\lambda^2}}{16}\right]\right\}^n \tag{24}$$

The proof is complete.

Corollary 1: For a given sensor node S_i, its own redundancy coverage satisfies the following condition.
$p(s_i) = 1 - \prod_{i=1}^{s_i \in S}\left(1 + \frac{\lambda\sqrt{4-\lambda^2}}{2\pi} - \frac{2\arccos\frac{\lambda}{2}}{\pi}\right)$
Proof: As shown in Figure 3, we assume that the distance between sensor nodes B and C is λ times than the sensing radius, the following relationship can be obtained.

$$\|S_B - S_C\| = \lambda r \tag{25}$$

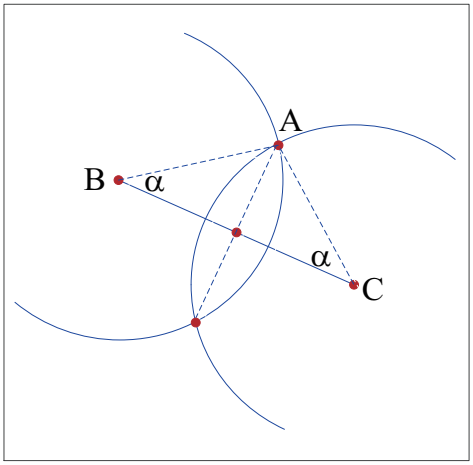


Fig. 3. Schematic diagram of redundant coverage of any adjacent sensor node

According to the trigonometric formula, we can obtain the following relationship.

$$\alpha = \angle ABC = \angle ACB = \arccos \frac{\lambda}{2} \tag{26}$$

The area of the intersection of sensor node B and C is shown as follows.

$$A_{(s_B, s_A)} = 2\alpha r^2 - \lambda r^2 \sin \alpha \tag{27}$$

The calculation for formula (27) can be obtained as follows.

$$A_{(s_B, s_A)} = 2r^2 \arccos \frac{\lambda}{2} - \frac{\lambda r^2 \sqrt{4 - \lambda^2}}{2} \tag{28}$$

In the monitoring area, the probability of any point being covered by s_i is as follows.

$$p_i = \frac{A_{(s_B, s_A)}}{\pi r^2} = \frac{2r^2 \arccos \frac{\lambda}{2} - \frac{\lambda r^2 \sqrt{4-\lambda^2}}{2}}{\pi r^2} = \frac{4 \arccos \frac{\lambda}{2} - \lambda \sqrt{4 - \lambda^2}}{2\pi} \tag{29}$$

In the monitoring area, the redundant coverage rate of a certain point covered by the sensor node s_i is as follows.

$$p\left(s_i\right) = 1 - \prod_{i=1}^{s_i \in S} \left(1 - p_i\right) = 1 - \prod_{i=1}^{s_i \in S} \left(1 - \frac{4 \arccos \frac{\lambda}{2} - \lambda \sqrt{4 - \lambda^2}}{2\pi}\right) \tag{30}$$

Simplifying formula (30) can be obtained as follows.

$$p\left(s_i\right)=1-\prod_{i=1}^{s_i\in S}\left(1+\frac{\lambda\sqrt{4-\lambda^2}}{2\pi}-\frac{2\arccos\frac{\lambda}{2}}{\pi}\right) \tag{31}$$

The proof is complete.

The purpose of sensor network coverage optimization is mainly embodied in the premise of ensuring a certain connectivity coverage rate, it uses the least sensor nodes to complete the largest area coverage in the monitoring area, thereby it will achieve the real-time network communication and maximizing the network life time.

In the actual application process, the location information of the mobile target node within the sensing range of the sensor node is often collected, analyzed and calculated in real time, which requires the sensor node to accurately cover the mobile target node, quantitative analysis of the sensor node coverage expectation and network coverage.

Theorem 4 is based on the research of Theorem 3, with the help of probability related knowledge, it derive and calculate the expected value of sensor node coverage.

Theorem 4: In the square monitoring area, the expected coverage of the sensor nodes is not greater than $E(X)=NP(n)$, where N is the number of sensor nodes and n is the number of working sensor nodes.

Proof. In the square monitoring area, n is the number of working sensor nodes, and $N-n$ is the number of sleeping sensor nodes. Therefore, the random variable leastways satisfies $((N-n),P(n))$ quadratic distribution .

$$P\left\{X=n\right\}=\begin{bmatrix}N-n\\n\end{bmatrix}P^n\left(n\right)\left[1-P\left(n\right)\right]^{N-n} \tag{32}$$

$$E\left(X\right)=\sum\nolimits_{n=1}^{N}nP\left\{X=n\right\}=\sum\nolimits_{n=1}^{N}n\begin{bmatrix}N\\n\end{bmatrix}P^n\left(n\right)\left[1-P\left(n\right)\right]^{N-n} \tag{33}$$

$$\sum\nolimits_{n=1}^{N}n\begin{bmatrix}N\\n\end{bmatrix}P^n\left(n\right)\left[1-P\left(n\right)\right]^{N-n}=\sum\nolimits_{n=1}^{N}\frac{nN!}{n!\left(N-n\right)!}P^n\left(n\right)\left[1-P\left(n\right)\right]^{N-n} \tag{34}$$

After calculation, we can obtain the following formula.

$$\begin{array}{l}\sum_{n=1}^{N}\frac{NP(n)(N-1)!}{(n-1)![(N-1)-(n-1)!]}\times P^{n-1}\left(n\right)\left[1-P\left(n\right)\right]^{N-n}\\=NP\left(n\right)\sum_{n=1}^{N}\frac{(N-1)!}{(n-1)!(N-n)!}\times P^{n-1}\left(n\right)\left[1-P\left(n\right)\right]^{N-n}\end{array} \tag{35}$$

Simplify formula (35) we can get formula (36).

$$NP\left(n\right)\left[P\left(n\right)+1-P\left(n\right)\right]^{N-1}=NP\left(n\right) \tag{36}$$

The proof is complete.

4.2. Analysis for MS-MEC Algorithm

The coverage process of the MS-MEC algorithm is covering moving targets with multiple types and related dimensions.

Algorithm 1: Triangulation algorithm

Step1: Calculating the difference vector between any sensor node s_i and s_j in the sensor network.

$$\overrightarrow{s_i s_j} = \overrightarrow{s_i} - \overrightarrow{s_j} \tag{37}$$

Step2: Calculating the equation and direction vector of the perpendicular line connecting the two nodes s_i and s_j, passing through the perpendicular point and perpendicular to the line connecting s_i and s_j.

$$\begin{cases} m_{ij} = s_j + \dfrac{\overrightarrow{s_i s_j}}{2} \\ l_{ij} = \mathrm{e}^{-i\frac{\pi}{2}} \overrightarrow{s_i s_j} \end{cases} \tag{38}$$

Step3: Repeating Step1 and Step2, calculating the vertical equation and direction vector of the third sensor node s_k in the sensor network, at the same time, calculating the intersection point c coordinate of $s_i s_j$ and $s_i s_k$.

$$\begin{cases} m_{ij} = s_j + \dfrac{\overrightarrow{s_i s_j}}{2} \\ l_{ij} = \mathrm{e}^{-i\frac{\pi}{2}} \overrightarrow{s_i s_j} \end{cases} \tag{39}$$

Step4: Calculating the distance among c and all sensor nodes.

$$c = \frac{m_{ij} \wedge l_{ij}}{l_{ij} \wedge l_{ik}} l_{ik} + \frac{m_{ik} \wedge l_{ik}}{l_{ik} \wedge l_{ij}} l_{ij} \tag{40}$$

Step5: If the distance between any sensor node and c is greater than or equal to the distance between node s_i and s_j, a set of triangulation graphs are formed. Otherwise, the algorithm ends.

Step6: Repeating Step1 to Step5 until all sensor nodes are traversed.

For the MS-MEC algorithm, in the initial stage, the sensor node and its neighbor nodes perform information interaction operations, and cooperate to complete the election process of candidate nodes. The process is as follows. (1) In the initial stage of the algorithm, all sensor nodes are in a ready state, and a random parameter δ is set. The random parameter δ is used as the logical basis for randomly determining the distribution of a certain sensor node, and then generates M different groups working nodes and spread Hello messages within the sensing range. The message includes the sensor node ID, location information, and coverage. In order to avoid repeated selection of candidate nodes, we introduce the Back-off Mechanism (BM), and set a random response time $T_{waiting}$ for each sensor node, $T_{waiting} \in [0, T]$ where T is the unit period. If the sensor node does not receive the Hello message within the specified time $T_{waiting}$, it will change its state to the candidate state. If the sensor node receives the Hello message from its neighbor node within the specified time $T_{waiting}$, it will use Definition 4 and Theorem 3 to calculate the coverage $T_{waiting}$ of the sensor node to the location of the target node. (2) After a certain

period T, all candidate nodes broadcast a notify message in a flooding manner, which includes coverage, node remaining energy, ID information, and node status information. If the energy of the candidate node is less than the threshold E_{min}, it will turn itself into a sleep state, where $E_{min}$is the threshold, and its setting formula is as follows.

$$E_{min} = \max\left(E_{tr} + E_{rx}\right) \tag{41}$$

Where E_{tr} and E_{rx} can be obtained as follows.

$$E_{tr} = E_{elec}\left(k\right) + E_{amp}\left(k, d\right) = \begin{cases} kE_{elec} + k\varepsilon f_s d^2 d < d_0 \\ kE_{elec} + k\varepsilon_{amp} d^4 d \geq d_0 \end{cases} \tag{42}$$

$$E_{rx}\left(k\right) = kE_{elec} \tag{43}$$

where E_{elce} represents the energy parameter consumed by the transceiver controller, εf_s represents the message energy parameter of the free space model, ε_{amp} represents the energy consumption parameter of the multi-channel attenuation model, d_0 is a constant, which depends on the actual application environment as shown in formula (43).

$$d_0 = \sqrt{\frac{\varepsilon f_s}{\varepsilon_{amp}}} \tag{44}$$

Algorithm 2: MS-MEC algorithm
Step1: Determining whether any sensor node can receive notify messages correctly. If it receives correctly, the sensor node is written into the candidate linked list N-Set, otherwise, the sensor node is not placed in the candidate linked list N-Set.
Step2: During the operation of the sensor node, if it receives an action message from the candidate node, it will update the state of the candidate node in the N-Set linked list, if it does not receive an action message, then enter Step 3.
Step3: If N-Set is non-empty, the set of sensor nodes in the linked list will be sorted according to the energy level. When the effective coverage rate of the target node is satisfied in advance, the sensor node with the highest energy sum will broadcast the action message and announce the node as working node. If the current coverage is not satisfied, the node with the second highest energy is selected; the above operation is repeated until a suitable sensor node is found, otherwise, the algorithm ends.
Step4: Determining whether the sensor node's perception probability $p(s)$ is greater than or equal to the threshold p_{th}, if it is greater than or equal to p_{th}, calculating the perception probability according to formula (44). If it is less than the threshold, we will continue to search the N-Set linked list to meet the conditions sensor node.

$$p\left(s\right) = 1 - \prod_{i=1}^{N-Set}\left(1 - p\left(s_i, s_j\right)\right) \tag{45}$$

Step5: If the N-Set linked list is empty, the sensor node will compete with the perceptual probability $p(s)$ until the entire N-Set linked list is traversed, in this case, if the N-Set

linked list is still empty, the algorithm jumps to Step2.
Step6: When a certain period ends, determining the status information of all sensor nodes, if it is a candidate or ready state, its sensor nodes will change to the sleep state.

4.3. MS-MEC Algorithm Complexity Analysis

In the worst state of the MS-MEC algorithm, the complexity of each sensor node is $O(1)$ when exchanging messages, and in a network with n sensor nodes, the complexity is $O(n)$. As far as each cycle is concerned, in order to make the distribution of randomly deployed sensor nodes more even, within the range of broadcasting Hello messages, the complexity of enabling neighbor nodes to receive this message is $O(1)$. In the loop iteration stage of n sensor nodes, if the number of loop iterations is m times, the maximum number of Hello messages broadcast by each candidate node is also m times, i.e., the complexity of exchanging messages is $O(m \times n)$.

5. Performance Evaluation

In the same simulation environment, the MS-MEC algorithm under different parameters is compared with other algorithms in three aspects, the number of sensor nodes, coverage and life time. Three different network monitoring areas are used: 100m $\times$ 100m, 300m $\times$ 300m, and 500m $\times$ 500m. The wireless sensor network is composed of 1200 nodes with a sensing radius of 10m, the initial energy of the sensor node is 5J, when the remaining energy of the sensor node is less than 0.005J, the sensor node can be considered as invalid.

5.1. Comparison Experiment of the Number of Sensor Nodes

In order to better verify the effectiveness of the MS-MEC algorithm in this paper, this paper uses different monitoring areas, takes the number of sensor nodes as the research background and compares with literature [24] (EMC-SC) and literature [28] (DCS-NC) and literature [40] (DR-ACO). Figure 4 to 6 show the difference between the number of sensor nodes and the number of working nodes of the MS-MEC algorithm, the DR-ACO algorithm, and the DCS-NC algorithm under different λ parameters and different time rounds in a monitoring area of 100m $\times$ 100m. Figure 7 to 9 show the difference between the number of sensor nodes and the number of working nodes in the MS-MEC algorithm, DR-ACO algorithm, and DCS-NC algorithm under different λ parameters and different time rounds in a monitoring area of 300m $\times$ 300m. Figure 10 to 12 show the difference between the number of sensor nodes and the number of working nodes of the MS-MEC algorithm, the DR-ACO algorithm, and the DCS-NC algorithm under different λ parameters and different time rounds in a 500m $\times$ 500m monitoring area.

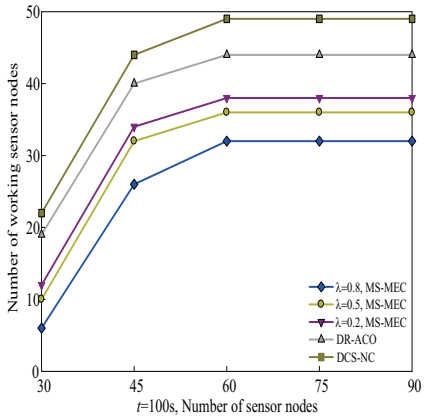


Fig. 4. 100m×100m,(t=100s) the number of nodes and working nodes comparison between among the three algorithms

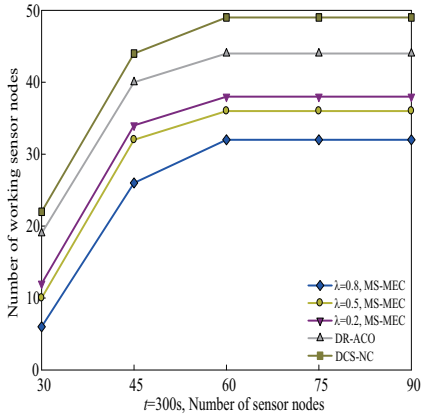


Fig. 5. 100m×100m(t=300s), the number of nodes and working nodes comparison between among the three algorithms

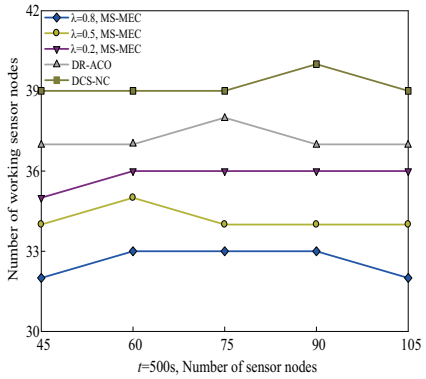


Fig. 6. 100m×100m(t=500s), the number of nodes and working nodes comparison between among the three algorithms

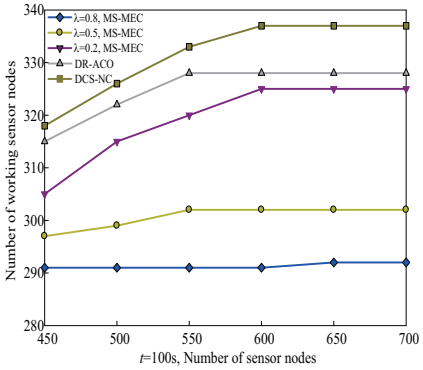


Fig. 7. 300m×300m(t=100s), the number of nodes and working nodes comparison between among the three algorithms

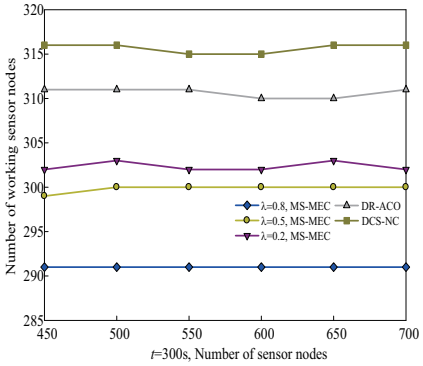


Fig. 8. 300m×300m(t=300s), the number of nodes and working nodes comparison between among the three algorithms

The MS-MEC algorithm is an optimized coverage algorithm with energy balance and migration strategy. The edge computing model effectively calculates the network coverage, node remaining energy and network life time and other attributes. In this paper, the MS-MEC algorithm uses the sensor node round mechanism to complete the effective coverage of the monitoring area, while DR-ACO uses the ant colony optimization algorithm to effectively cover the monitoring area. The algorithm only considers the optimization process and the one-way path. The coverage mechanism simply completes the coverage of a single target node without considering the problem of mobile multi-target coverage; the DCS-NC algorithm uses full coverage of the monitoring area to complete the coverage of the mobile target node without considering the sensor node energy consumption problem. It can be seen from Figure 4 to Figure 6 that in the 100m × 100m monitoring area, as the number of sensor nodes increases, the number of working nodes of the

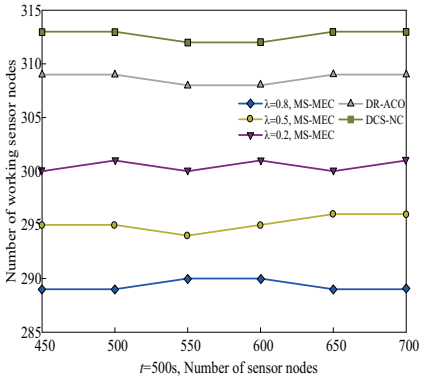


Fig. 9. 300m×300m(t=500s), the number of nodes and working nodes comparison between among the three algorithms

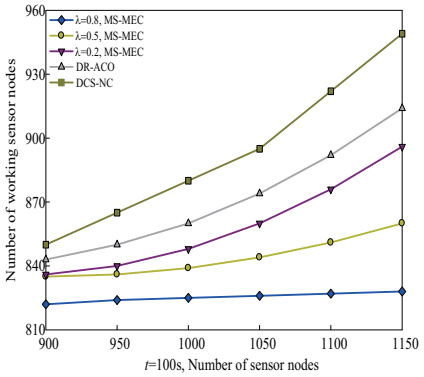


Fig. 10. 500m×500m(t=100s), the number of nodes and working nodes comparison between among the three algorithms

three algorithms also increases, but the increase in the MS-MEC algorithm in this paper is much smaller than the other two algorithms. When t=100s, λ=0.8, the number of working nodes required by the MS-MEC algorithm in this paper is 32, the increase value is the smallest, and the number of sensor nodes in working state is also the smallest. Secondly, when t=100s, λ =0.5, the number of working nodes required by the MS-MEC algorithm in this paper is 36, which means that the network energy balance is achieved. Thirdly, when t=100s and λ=0.2, the number of working nodes required by the MS-MEC algorithm in this paper is 38, which means that the network energy balance is achieved. But under the same conditions, the DR-ACO algorithm and the DCS-NC algorithm require 44 and 49 sensor nodes to achieve network equilibrium. In a 100m × 100m monitoring area, the average number of sensor nodes of the MS-MEC algorithm is 75.91% of the other two algorithms. At t=300s and t=500s, the average number of sensor nodes used in the MS-MEC algorithm is 86.65% and 89.47% of the other two algorithms. In terms of suppressing the number of nodes, the MS-MEC algorithm is within a 100m × 100m mon-

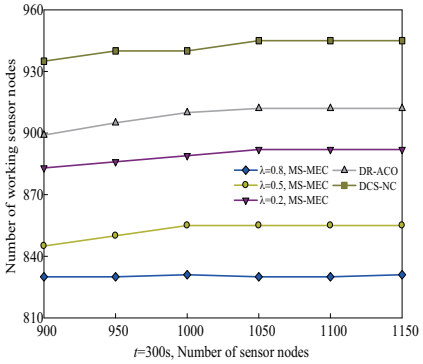


Fig. 11. 500m×500m(t=300s), the number of nodes and working nodes comparison between among the three algorithms

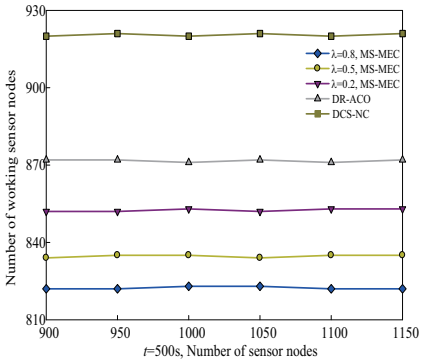


Fig. 12. 500m×500m(t=500s), the number of nodes and working nodes comparison between among the three algorithms

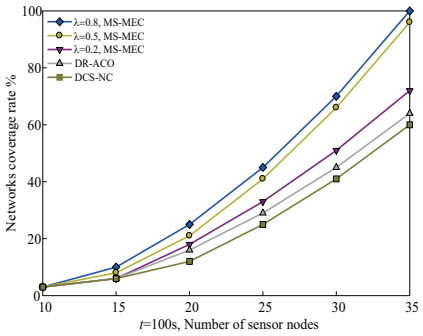


Fig. 13. 100m×100m,(t=100s), the comparison of network coverage rate with the three algorithms

itoring area, and the average number of sensor nodes used is 15.99% less than the other two algorithms. According to the above analysis process, the average number of sensor nodes used in the MS-MEC algorithm is 94% and 92.78% of the other two algorithms. In terms of suppressing the number of nodes, the average number of sensor nodes used in the MS-MEC algorithm is 6% and 7.22% less than the other two algorithms in the monitoring area of 300m ×300m and 500m × 500m. Based on the above analysis, the average number of working sensor nodes of the MS-MEC algorithm is 9.74% less than the other two algorithms under different parameters, different monitoring areas and different time rounds.

5.2. Coverage Quality Comparison Experiment

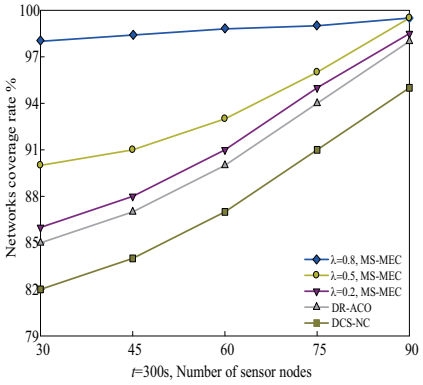


Fig. 14. 100m×100m(t=300s), the comparison of network coverage rate with the three algorithms

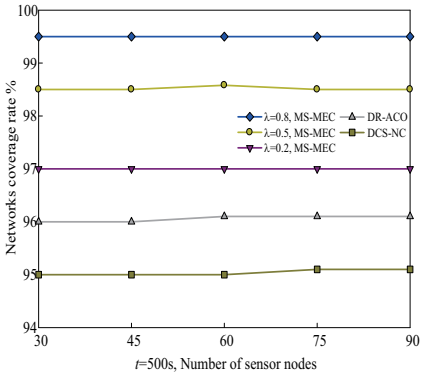


Fig. 15. 100m×100m(t=500s), the comparison of network coverage rate with the three algorithms

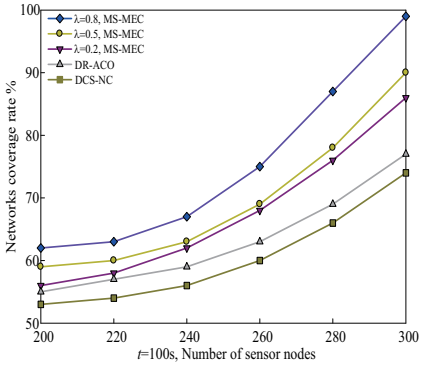


Fig. 16. 300m×300m(t=100s), the comparison of network coverage rate with the three algorithms

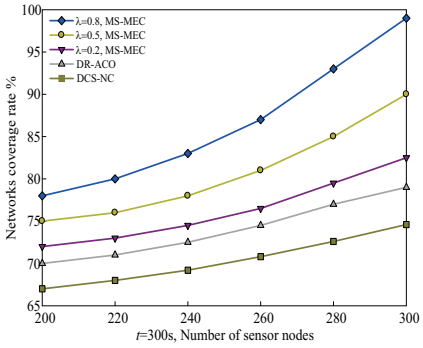


Fig. 17. 300m×300m(t=300s),the comparison of network coverage rate with the three algorithms

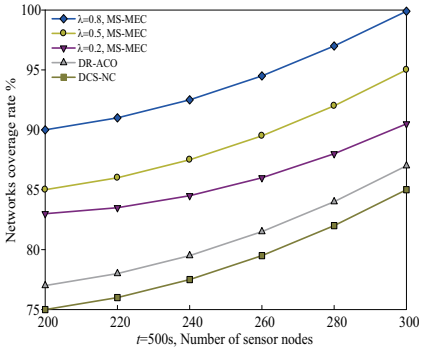


Fig. 18. 300m×300m(t=500s), the comparison of network coverage rate with the three algorithms

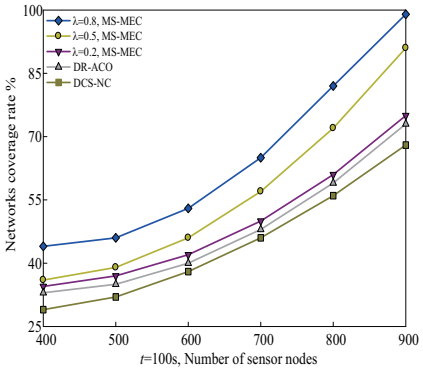


Fig. 19. 500m×500m(t=100s),the comparison of network coverage rate with the three algorithms

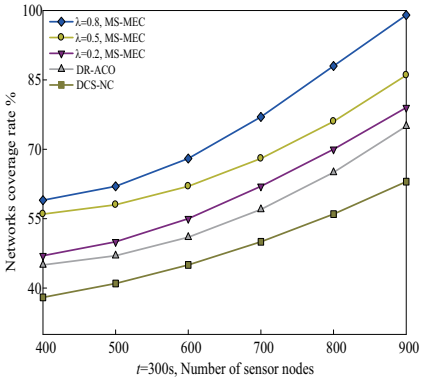


Fig. 20. 500m×500m(t=300s), the comparison of network coverage rate with the three algorithms

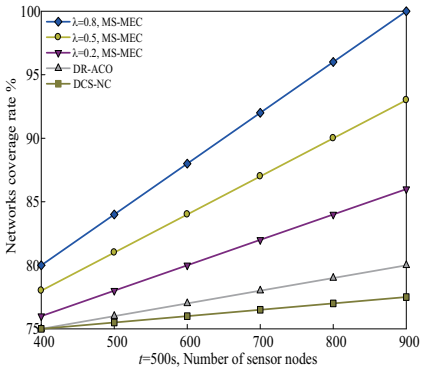


Fig. 21. 500m×500m(t=500s), the comparison of network coverage rate with the three algorithms

Figure 13 to Figure 21 respectively show the comparison of the network coverage rate and the number of sensor nodes of the three algorithms in different monitoring areas. We take Figure 16 to Figure 18 as an example for analysis. From the above three figures, it can be seen that with the increase in the number of sensor nodes, the network coverage rate of the three algorithms have increased, but the increase of MS-MEC algorithm is higher than that of DR-ACO algorithm and DCS-NC algorithm. When t=300s and the number of sensor nodes is 260, the MS-MEC algorithm network coverage is λ=0.8, p=87%, λ=0.5 p=81% and λ=0.2,p=76.5% . The network coverage rate of the DR-ACO algorithm is p=74.5%, the network coverage rate of the DCS-NC algorithm is p=70.8%, at this time, the average network coverage rate of the MS-MEC algorithm is 8.85% higher than the other two algorithms. When the number of sensor nodes is 300, the network coverage rate of the MS-MEC algorithm is λ=0.8,p=99%, λ=0.5, p=90% and λ=0.2,p=82.5%, and the

network coverage rate of DR-ACO algorithm is p=79%, and the network coverage rate of the DCS-NC algorithm is p=74.6%. The average network coverage rate of the MS-MEC algorithm is 13.7% higher than the other two algorithms. The main reason is as follows. The MS-MEC algorithm effectively controls the network coverage by adjusting the parameters, thereby increasing the effective coverage of the local monitoring area, while the DR-ACO algorithm and the DCS-NC algorithm only use the full coverage characteristics of sensor nodes to cover effectively the entire monitoring area. The DR-ACO algorithm and the DCS-NC algorithm do not achieve effective coverage through certain parameter control, and on the other hand, in terms of the coverage, the two algorithms do not consider the entire network energy consumption problem, which increases the rapid consumption of sensor node energy. For a large-scale monitoring area of 500m $\times$ 500m, the comparison diagram of network coverage rate and sensor nodes given is shown in Figure 19 to 21. We take Figure 21 as an example for analysis. In Figure 21, we can see that the network coverage rate increases with the increase of the number of sensor nodes. The increase of network coverage rate of the MS-MEC algorithm is higher than that of the other two algorithms. When the number of sensor nodes is 600, the coverage rate of the MS-MEC algorithm network is λ=0.8, p=88%, λ=0.5,p=84% and λ=0.2,p=80%, and the DR-ACO The network coverage rate of the algorithm is p=77%, the network coverage rate of the DCS-NC algorithm is p=76%. However the average network coverage rate of the MS-MEC algorithm is 7.5% higher than the other two algorithms. When the number of sensor nodes is 900, the coverage of the MS-MEC algorithm network is λ=0.8, p=99%, λ=0.5, p=93%, λ=0.2, p=86%, and the network coverage rate of the ACO algorithm is p=80%, the network coverage rate of the DCS-NC algorithm is p=77.5%, however, the average network coverage rate of the MS-MEC algorithm is 13.91% higher than the other two algorithms. Based on the above analysis, the coverage rate of the MS-MEC algorithm network is 9.92% higher than that of the other two algorithms.

6. Conclusion

This article mainly conducts research from three aspects, the number of nodes in the sensor network, network coverage, and network energy consumption. The proposed MEC-MS algorithm uses the network coverage model to give the distance measurement judgment conditions, uses the probability-related theoretical knowledge to analyze the necessary conditions for improving the coverage quality and the preconditions for the existence of the redundant coverage of adjacent redundant coverage nodes. On the basis of this analysis, using the preconditions of redundant coverage, the calculation process of the sensor node's own redundant coverage and the calculation method of redundant node coverage expectations are given. The experimental results show that the MEC-MS algorithm is superior to the other two algorithms in terms of the number of nodes, network coverage and network energy consumption.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (No.62076096,61931016,61771015); Science and Technology Research Project of Henan Province (No. 212102210374); Natural Science Foundation of Henan Province (No. 202300410286); Department of Education Key Funding for Natural Science Foundation of Henan Province Education (No. 19A520006, 20A520027).

References

1. Donta, P. K., R.B.S.A.T.: Data collection and path determination strategies for mobile sink in 3d wsns. IEEE Sensors Journal 20(4), 2224–2233 (2020)
2. erma, A., K.S.G.P.R.: Broadcast and reliable coverage based efficient recursive routing in large-scale wsns. Telecommunication Systems 75(1), 63–78 (2020)
3. Gao, X. F., C.Z.Y.P.J.P.: Energy efficient scheduling algorithm for sweep coverage in mobile sensor networks. IEEE Transactions on Mobile Computing 19(6), 1332–1345 (2020)
4. Harizan, S., K.P.: Coverage and connectivity aware energy efficient scheduling in target based wireless sensor networks: An improved genetic algorithm based approach. Wireless Networks 25(4), 1995–2011 (2019)
5. Huang, M. F., L.A.F.Z.M.: Multi working sets alternate covering scheme for continuous partial coverage in wsns. Peer-to-Peer Networking and Applications 12(3), 553–567 (2019)
6. Javan, B. A., M.H.M.H.: A learning automate-based algorithm to solve imbalanced k-coverage in visual sensor networks. Journal of Intelligent and Fuzzy Systems 39(3), 2817–2829 (2020)
7. Khalaf, O. I., A.G.M.S.B.M.: Optimization of wireless sensor network coverage using the bee algorithm. Journal of Information Science and Engineering 36(2), 377–386 (2020)
8. Khalifa, B., K.A.M.A.A.Z.: A coverage maintenance algorithm for mobile wsns with adjustable sensing range. IEEE Sensors Journal 20(3), 1582–1591 (2020)
9. Kim, J., Y.Y.: Sensor node activation using bat algorithm for connected target coverage in wsns. Sensors 20(13), 1–24 (2020)
10. Krishnan M., Rajagopal, V.R.S.: Performance evaluation of sensor deployment using optimization techniques and scheduling approach for k-coverage in wsns. Wireless Networks 24(3), 683–693 (2018)
11. Li, H., O.K.D.M.X.: Deep reinforcement scheduling for mobile crowd-sensing in fog computing. ACM Transactions on Internet Technology 16(5), 1623–1638 (2020)
12. Lin, Z., K.H.C.W.R.K.: Joint data collection and fusion using mobile sink in heterogeneous wireless sensor networks. IEEE Sensors Journal 21(2), 2364–2376 (2021)
13. Liu, Y., C.K.W.Y.C.L.: Node deployment for coverage in rechargeable wireless sensor networks. IEEE Transactions on Vehicular Technology 68(6), 6064–6073 (2019)
14. Liu, Q., H.P.L.W.G.J.: Intelligent route planning on large on large road networks with efficient and privacy. Journal of Parallel and Distributed Computing 133, 93–106 (2019)
15. Liu, Z. Z., L.S.N.: Data collection scheme based on expected networks coverage and cluster compressive sensing for wsns. Control and Decision 33(3), 422–430 (2018)
16. Liu, X. X., L.M.L.C.W.T.L.A.F.: Movement-based solutions to energy limitation in wireless sensor networks: State of the art and future trends. IEEE Network Early Access Article, 1–6 (2020)
17. Liu, X. X., Q.T.D.B.: Swarm intelligence-based rendezvous selection via edge computing for mobile sensor networks. IEEE Internet of Things Journal 7(10), 9471–9480 (2020)
18. Liu, X, X.L.P.H.L.T.: Objective-variable tour planning for mobile data collection in partitioned sensor networks. IEEE Transactions on Mobile Computing (Early Access), 1–1 (2020)
19. Njoya, A. N., A.A.A.N.A.M.: Hybrid wireless sensor deployment scheme with connectivity and coverage maintaining in wireless sensor networks. Wireless Personal Communications 112(3), 1893–1917 (2020)

20. Patel, D., J.A.: Improving area coverage with mobile node in wireless sensor networks. International Journal of Interdisciplinary Telecommunications and Networking 13(1), 36–48 (2021)
21. Qi, H. W., B.H.Y.R.H.: Automatic monitoring systems of energy-saving agricultural production equipment using wireless sensor networks. International Journal of Mechatronics and Applied Mechanics 2(8), 150–157 (2020)
22. Saadi, N., B.A.E.R.: Maximum lifetime target coverage in wireless sensor networks. Wireless Personal Communication 111(3), 1525–1543 (2020)
23. Singh, M.K.: Discovery of redundant free maximum disjoing set-k-covers for wsn life enhancement with evolutionary ensemble architecture. Evolutionary Intelligence 13(4), 611–630 (2020)
24. Singh, S., S.R.M.: Heuristic based coverage aware load balanced clustering in wsns and enablement of iot. International Journal of Information Technology and Web Engineering 13(2), 1–10 (2018)
25. Sun, Z. Y., L.L.X.X.X.F.: A novel node deployment assignment scheme with date association attributed in wireless sensor networks. Journal of Internet Technology 20(2), 509–520 (2019)
26. Sun, Z. Y., W.L.L.X.C.: An event-driven mechanism coverage algorithm based on sensing-cloud-computing in sensor networks. IEEE Access 7, 84668–84679 (2019)
27. Sun, Z. Y., X.X.F.W.T.: An optimized clustering communication protocol based on intelligent computing in information-centric internet of things. IEEE Access 7, 28238–28249 (2019)
28. Sun, Z, Y.L.Z.G.H.Y.: Mr-dfm: A multi-path routing algorithm based on data fusion mechanism in sensor networks. Computer Science and Information Systems 16(3), 867–890 (2019)
29. Sun, Z. Y., Z.Y.S.N.Y.L.: Casmoc: A novel complex alliance strategy with multi-objective optimization of coverage in wireless sensor networks. Wireless Networks 23(4), 1201–1222 (2017)
30. Sun, Z. Y., Z.G.Z.X.X.F.: Encp: A new energy-efficient nonlinear coverage control protocol in mobile sensor networks. EURASIP Journal of Wireless Communications and Networking 20(18), 1–15 (2018)
31. Wang, T., C.Z.H.W.S.: Privacy-enhanced data collection based on deep learning for internet of vehicles. IEEE Transactions on Industrial Informatics 16(10), 6663–6672 (2020)
32. Wang, J., J.C.W.K.H.J.: A mobile assisted coverage hole patching scheme based on particle swarm optimization for wsns. Cluster Computing 22, 1787–1795 (2019)
33. Wang, T., Z.D.C.S.B.: Bidirectional prediction- based underwater data collection protocol for end-edge- cloud orchestrated system. IEEE Transactions on Industrial Informatics 16(7), 4791–4799 (2020)
34. Wu, Y. K., H.H.Y.W.Q.: A risk defense method based on microscopic state prediction with partial information observations in social. Journal of Parallel and Distributed Computing 31, 189–199 (2019)
35. Xu, X, H.D.Z.X.S.A.X.G.T.: Connected target-probability coverage in wsns with directional probabilistic sensors. IEEE Systems Journal 14(3), 3399–3409 (2020)
36. Xu, H., W.B.L.S.J.: An algorithm for calculating coverage rate of wsns based on geometry decomposition approach. Peer-to-Peer Networking and Applications 12(3), 568–576 (2019)
37. Xu, Y. L., Y.Y.D.Z.H.: A fast two-objective differential evolution for the two-objective coverage problem of wsns. Memetic Computing 11(1), 89–107 (2019)
38. Yang, G. S., L.T.T.H.X.Y.: Global and local reliability-based routing protocol for wireless sensor networks. IEEE Internet of Thing Journal 6(2), 3620–3632 (2019)
39. Zanaj, E., G.E.Z.B.: Customizable hierarchical wireless sensor networks based on genetic algorithm. International Journal of Innovative Computing, Information and Control 16(5), 1623–1638 (2020)
40. Zhang, D. C., S.W.E.R.: A coveragee and obstacle-aware clustering protocol for wireless sensor networks in 3d terrain. Computer Communications 146, 48–54 (2019)
41. Zhang, Y. F., W.Y.C.: A novel energy-aware bio-inspired clustering scheme for iot communication. Journal of Ambient Intelligence and Humanized Computing 11(10), 4239–4248 (2020)

42. Zhang, Y.: Coverage optimization and simulation of wireless sensor networks based on particle swarm optimization. International Journal of Wireless Information Networks 27(2), 307–316 (2020)
43. Zorbas, D., D.C.: Connected coverage in wsns based on critical targets. Computer Networks 55(6), 1412–1425 (2011)

Zeyu Sun was born in Changchun. He received the B.S. degree in computer science and technology from the Henan University of Science and Technology, in 2003, the M.S. degree from Lanzhou University, in 2010, and the Ph.D. degree form Xi an Jiaotong University, in 2017. He is currently pursuing the second Ph.D. degree in Xidian University, Xi an, China. He is currently a Professor with the School of Computer and Information Engineering, Luoyang Institute of Science and Technology, Luoyang, Henan, China. His research interests include wireless sensor networks, mobile computing, Internet of Things, collaborative computing and fog computing.

Guisheng Liao (Senior Member, IEEE; Corresponding Author) was born in Guilin. He received the B.S. degree in mathematics from Guangxi University, Guangxi, China, in 1985, and the M.S. degree in computer software and Ph. D. degree in signal and information processing from Xidian University, Xi an, China, in 1990 and 1992, respectively. He is currently a Professor with the School of Electronic Engineering and a Yangtze River Scholars Distinguished Professor with the National Laboratory of Radar Signal Processing, Xidian University. His research interests include array signal processing, space-time adaptive processing, collaborative computing and edge computing.

Cao Zeng received the B.E. degree in electronic engineering, the M.S. degree in information and communication engineering, and the Ph.D. degree from Xidian University, Xi an, China, in 2001, 2004, and 2008, respectively. He is currently an Assistant Professor with the Department of Electronic Engineering, Xidian University. His research interests include array signal processing, multichannel moving target indication, and real-time system design and development of signal processing.

Zhiguo Lv received the B.S. degree in applied electronic technology form Henan Normal University, in 2000, and the M.S. degree in communication and information system from the Guilin University of Electronic Technology, in 2008 , and the Ph.D. degree from Xidian University, in 2019. He is a Lecturer with the Luoyang Institute of Science and Technology. His research interests include compressive sensing, wireless sensor networks and MIMO systems.

Chen Xu received the M.S. degree from Lanzhou Jiaotong University, in 2010. He is currently pursuing the Ph.D. degree in Tongji University, Shanghai, China. He is a Lecturer with the Shanghai Institute of Technology. His research interests include mobile computing, intelligent transportation systems, and wireless sensor networks.

Received: September 30, 2021; Accepted: January 27, 2022.

Computer Science and Information Systems 19(2):857–886 https://doi.org/10.2298/CSIS211009007R

Recent Advancements in Privacy-aware Protocols of Source Location Privacy in Wireless Sensor Networks: a Survey

Pradeep Kumar Roy[1], Asis Kumar Tripathy[2], Sunil Kumar Singh[3], and Kuan-Ching Li[4,⋆]

[1] Department of Computer Science and Engineering
Indian Institute of Information Technology, Surat, India
pradeep.roy@iiitsurat.ac.in
[2] School of Information Technology and Engineering
Vellore Institute of Technology, Vellore, Tamil Nadu, India
asistripathy@gmail.com
[3] School of Computer Science and Engineering
VIT-AP University, Near Vijaywada, Andhra Pradesh, India
sksingh.cse@gmail.com
[4] Department of Computer Science and Information Engineering (CSIE),
Providence University, Taiwan
kuancli@gm.pu.edu.tw

Abstract. This review article summarises the protocols proposed in recent researches to secure location information in Wireless Sensor Networks (WSNs). Due to their lightweightness and easy to deploy properties, WSNs are widely used in numerous object tracking and monitoring applications. Due to such, source location privacy attracts the researchers and hence continuously enhances its improvement. Though, this privacy breach is not acceptable for WSNs, as it may reveal some critical information that is harmful. The SLP issue on WSN attracted researchers a lot, and hence a number of solutions are provided for it. However, an up-to-date survey does not exist for the same. To fill this gap, in this article, we summarize different approaches proposed in the last years to preserve location privacy. We first discuss the different privacy characteristics in WSNs, a detailed overview of the proposed protocols and their limitations, and discussions of solutions for the adversaries' capabilities in WSNs. Then the future research directions in this area are discussed. This review work may support researchers identifying the new research area in location privacy of wireless sensor networks.

Keywords: Wireless sensor networks, Source location privacy, Fake source, Phantom routing, Security,

1. Introduction

Wireless sensor networks (WSNs) are typically composed of sensor nodes with limited power, memory, computational capabilities, and communication resources. The sizes of these sensor nodes are tiny, with limited computing and processing resources, and are

⋆ Corresponding author

cheaper than conventional sensors. However, WSNs provide potentially low-cost solutions to multiple issues in both civilian and military applications, along with target monitoring, battlefield surveillance, health care, environmental monitoring, traffic regulation, and wildfire detection[1,72]. In recent years, WSNs have attracted global attention, particularly with the proliferation of Micro-Electro-Mechanical Systems (MEMS) technology that helps a lot in the intelligent sensors development process [2]. These sensor nodes can sense, measure, and collect data from the environment and transmit the sensed information to the user based on routing techniques [3,73,71]. Researchers focused on the main characteristics of the WSNs, such as the sensors' energy conservation, their computational power, and the resource constraints. However, addressing the privacy issues in the WSNs are received very little attention [68]. Privacy in WSN refers to private information such as monitoring messages, object tracking messages, and others transmitted over the network. For example, a patient's blood pressure, sugar level, and various critical symptoms are usually essential concerns of privacy that need to be secured while transmitting this information to a faraway health centre or doctor's office using the WSNs. Privacy concerns may also arise beyond the information content and may include knowledge about context information that consists of a sensor's location starting information communication.

This paper focused on summarizing the recent works on monitoring and tracking applications with wireless sensor networks. The applications include the monitoring of doctors and patients movement in the hospital and wildlife tracking [4]. The sensor network is used for monitoring the objects and tracking their movements. Figure 1 shows the issue of SLP in object tracking, an adversary sitting near to base station and listing all incoming messages. Further, by following the route of the incoming message, they can reach the origin of the message to trap the object. The object might be a human being, a vehicle, or an animal. When the sensor nodes sense the object's presence, it passes the sensed information to the nearby one or more sinks [5]. Further, the collected data may be forwarded to the server or allows manual extraction to extract the information. Providing the confidentiality of the communication between nodes for message exchanging does not help to secure the source's location. SLP needs more than concealment of message exchange between the nodes in the network. Also, the confidentiality of a WSN message is part of another privacy policy called content privacy[5]. The main focus of content privacy is to provide the integrity, confidentiality, and availability of the message in WSNs. In contrast, SLP and sink location privacy are part of context privacy that aims to hide the contextual information in WSNs [6]. The SLP in WSN consisting three main components apart from the sensor node, including- Source, Sink, and Adversary. The sensitive information is originated from the source node in the network and is delivered to the base station using suitable network protocol in multiple packets. The number of packets depends upon the size of the information. Further, an adversary sitting near the base station starts following the route of incoming packets to find the origin of the message in the network.

Conti et al. [7] provided an extensive survey on SLP; however, many latest protocols were proposed in recent years that need to be discussed. Li et al. [8] explained the types of privacy in WSN. Aivaloglou et al. [9] provide a survey in which only discussed half of the solutions that were already discussed in [8]. As shown in Table 1, the existing surveys were not up to date and lacked future research directions. To fill this gap, this article summarizes the solutions that the researchers proposed to date. For this survey, we have collected the research articles from the different libraries such as *Elsevier*, *IEEE*, *ACM*,

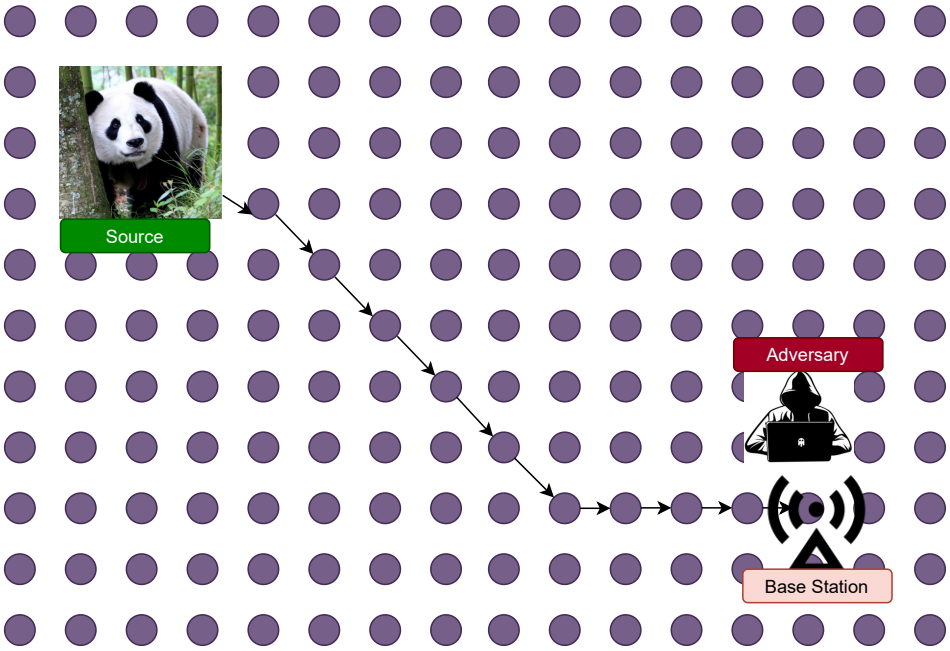


Fig. 1. Scenario of object tracking by following the incoming message to the base station

Springer, and *Scopus* using the keywords like: "source location privacy", "sink location privacy", "fake source", "phantom routing", and "privacy" published during year 2004 to April 2021. The collected articles were re-evaluated to check their belonging in the proposed aim and scope. Articles that were not fit was not added to this work. Our major contributions include the following:

1. Collected and organized high-quality research articles from various sources.
2. Discussed in brief, the privacy issues in WSNs.
3. Explored the different models and architectures that were used to provide the SLP in WSNs.
4. Briefly explains the existing models' limitations and provides future research directions.

The rest of the article is organized as follows: Section 2 discusses the background of location privacy. The solution to the SLP using fake source and phantom routing is shown in Section 3. In Section 4, the SLP challenges for WSN is discussed. The future research direction is detailed in Section 5. In Section 6, we conclude this work with limitations in the existing protocols.

2. Background

In recent year researcher put a lot of attention to preserve location privacy in WSNs [10]. In WSNs, privacy issues are mainly categorized in two parts: data privacy and context

privacy, as shown in Fig. 2. This section focused on the different concepts proposed by the researchers for SLP and adversary capabilities. In data privacy, the security mechanism is

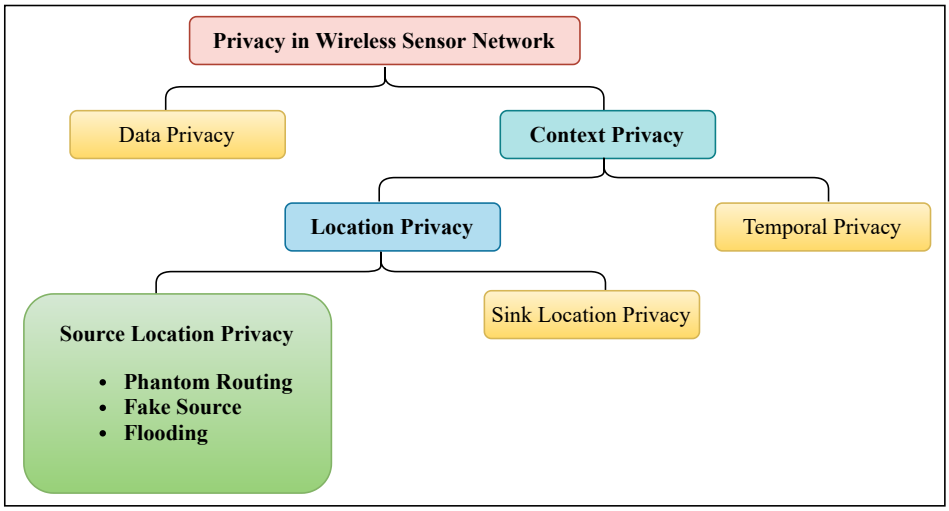


Fig. 2. Privacy issues in Wireless Sensor Networks

mainly implemented to provide security to the packets transmitted in WSNs. In context privacy, the objective is to provide privacy to context, such as the location of the sensor nodes.

2.1. Adversary Model

The main goal of any privacy-preserving protocol is to create confusion in the backtracking route of an adversary. Hence, the adversary has to spend more time in the wrong direction. This will increase the source node's safety period. Researchers have made the following assumption about an adversary to proposed the privacy protocols in WSN.

- An adversary is resource-rich as they have more storage and a more range of tracing power.
- An adversary is passive. They monitor the flow of traffic without making any detectable change.
- An adversary has a sectional antenna, through which they can predict the direction of incoming messages. Hence, they start backtracking the packets from the sink to reach the source.
- An adversary may store the visited node ID that helps avoid entering the loop if any loop is present or not visiting the same node again.
- An adversary knows the sink node location and silently stays there and waits for a message.

Table 1. List of the existing surveys on privacy models in WSN

Source	**Major Focus**	**Topic discussed**	**Remarks**
Conti et al. [7]	Source location privacy	overview of the solutions that provide source location privacy within a WSN	challenges are not provided, survey is very old.
Li et al. [8]	privacy-preserving techniques for WSNs	mainly discussed two privacy techniques data-oriented and context-oriented	more than 1 decade old and source location privacy not covered
Rios et al. [11]	Location privacy	analyse whether traditional communication systems are comfortable to the requirements of location privacy in sensor networks	Only communication related issues are discussed and survey is very old.
Jiang et al. [12]	Privacy models	mainly concern on privacy models to see their comparability and suitability analysis for different scenarios.	Only 5 years papers are considered in this survey.
Gupta and Prince [13]	Source location privacy	mainly deals on SLP but in random walk model.	Only random walk model related works are included with limited papers.
Jiang et al. [14]	Location privacy protection	This survey is classified into three categories i) source node's location privacy protection ii) sink nodes' location privacy protection, and iii) location privacy protection for both source and sink nodes	survey is good but it is mainly based on location privacy.
Our Survey	Source location privacy	source location privacy with adversary model discussed in detail.	Only focused on source location privacy.

- When a message arrives at the sink node, it predicts the direction of that incoming message with the help of section antenna and moves towards that node.
- After moving, the adversary wait for the next message for a fixed amount of time, called the observation period. If any message arrives during that time, then move again else return to the previous node location.
- The above procedure is continued until the adversary reaches the source node.

2.2. Network Model

The main issue with the WSNs is the network lifetime. Due to the limited battery power of the sensor devices, the network lifetime is one of the main research issues. To save the sensor node's energy and increase the network lifetime, a network may split into some clusters or grids [15]. Such a network is helpful to save the energy of the sensor node in large WSNs. However, the researcher proposed the solutions concerning the flat networks where all the sensor nodes are active and homogeneous. The nodes have the same battery power, processing capabilities, and storage capacity. In the network, nodes are deployed randomly to monitor the object and transmit it to the sink node using a multi-hop communication technique. An adversary is there to breach the privacy of the network. It is assumed that the adversary has more battery power, processing capabilities, and storage capacity as compared to the normal sensor nodes [5,70]. They may introduce some malicious nodes in the network, and hence they find out the locations of the source or sink node [16].

There are several works that have been done to preserve the source location privacy using the different approaches [17,18,19]. Among all the different techniques, the most effective technique is the fake source. Researchers have been proved that the protocol based on the fake source is more efficient to achieve better SLP. To the best of our knowledge, there is no updated review article published in the recent year that summarizes current researches. Hence, it is necessary to collect and summarize the research progress, highlight the limitations, and provide future research directions. The updated review may help new researchers in the domain to identify the gaps in ongoing research and proposed new frameworks.

2.3. Inclusion-Exclusion Method

As shown in Figure 3 and 4, the research on SLP was started in year 2004, then it continues. The total number of articles downloaded from the various digital libraries with the help of search keywords is 924. Many articles found duplicates and even not uses the sensor concepts, which was removed and left with 612 articles. The further manual screening was done from our side and excluded 473 articles, as they used the WSN concepts but not for preserving the SLP and hence did not fit our objective. From the remaining 139 articles, the non-English, or without proper simulation details, discusses only security issues but not privacy; not included adversary details are excluded and left with 82 articles.

3. Privacy Protocols with Fake Source and Phantom Routing

The privacy issues in WSN was first introduced by Ozturk et al. [17] with the help of the panda hunter game. They used four different concepts to preserve the SLP of the sensor

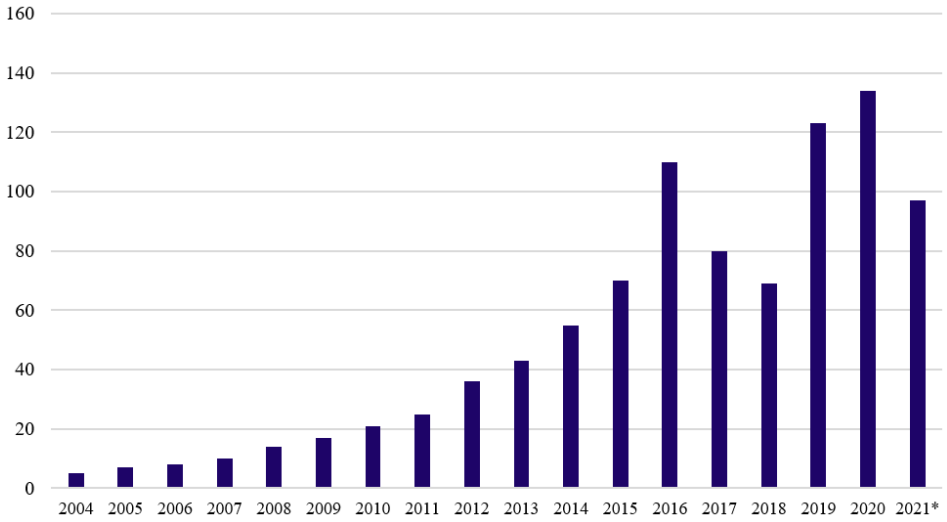


Fig. 3. Number of articles published for SLP in WSNs

node (i) Baseline flooding technique, (ii) Probabilistic flooding technique, (iii) Flooding with fake messages, and (iv) Phantom flooding.

In the Baseline flooding technique, all intermediate sensor node only transfer the message once. Whenever a sensor node receives a message from its neighbour, it checks first whether it is receiving the first time or not, if the first time, then forward, else discard the message. In *probabilistic flooding technique*, only the subset of sensor nodes will participate instead of all the sensor nodes. In this way, the network lifetime was improved. Each node forwards the packet with the dependency of forwarding probability p. The model's drawback was that the networks might be disconnected if the messages were lost while they were in the transit phase. The third approach to preserve the SLP was flooding with fake messages. To mislead the adversary, some fake sources were created on the network to flood the fake messages. Fake messages are similar to real ones. An adversary receives a fake message they cannot differentiate. As a result, they may lead to fake sources instead of the real source node. In the phantom flooding approach, the message delivers to the base station in two phases. First, the message passes up to h hops using either a random walk or directed random walk. Second, flooding technique is used to deliver it to the sink node as shown in Fig. 5.

Two grid-based SLP schemes, namely single phantom node SLP scheme (SPS) and dual phantom node SLP protection scheme (DPS) was proposed [20]. Here, the sink node helps the source node select the phantom node candidate set (PNCS). The source node randomly selects the fake source node from the PNCS. A location privacy mechanism based on fake source nodes was described to keep the source location secret from the global adversary [21,65,67]. Here, the adversary can see the entire network traffic in an energy-efficient manner. A two hierarchy shadow routing was proposed to checkmate the adversary [22]. The adversary gets two levels of obstruction during the enforcement of the traffic analysis attack. The two protocols, namely Two-level phantom with a pursue

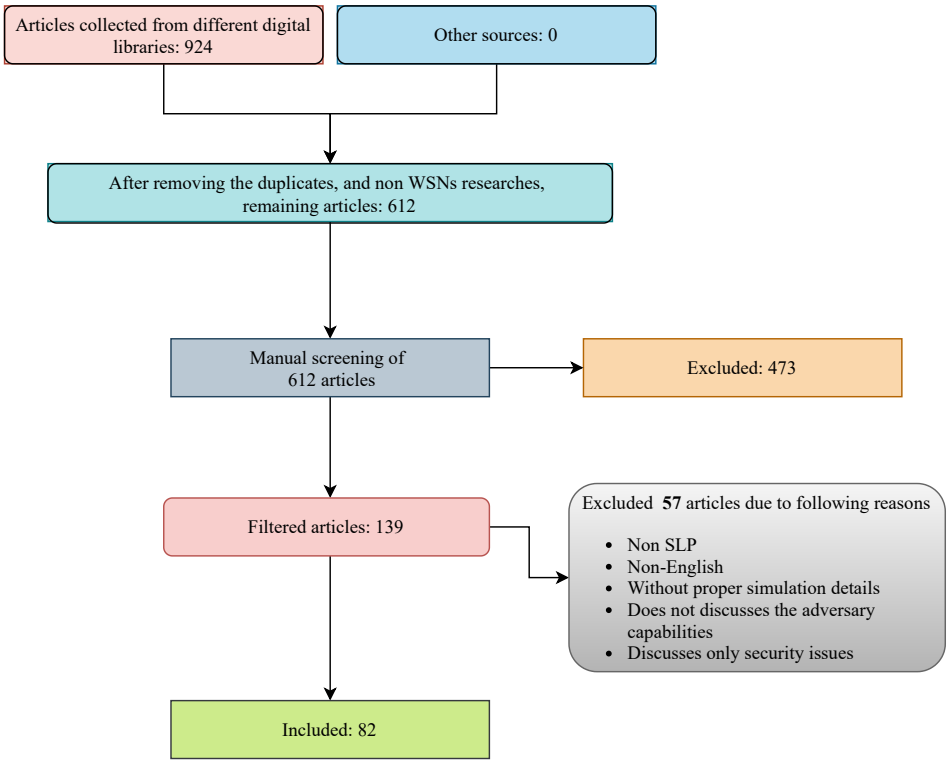


Fig. 4. Process of including and excluding the research articles

ring protocol (PhaP) and Two-level phantom with a backbone route protocol (PhaT), are described, which overcomes the drawbacks of the fake source routing schemes.

To improve the source node's safety period [6] modified the fake source routing technique and suggested a new protocol called phantom routing. In the fake source routing, the fake source node's position is very important because if the fake source node is situated between the real source and the base station, then an adversary may reach the real source while backtracking the messages. In Fig. 6 there are few fake source are shown, among these fake sources the choice of *f1* is not good whereas *f2, f3, f4, f5, f6* are better choice. Among these locations, if a fake source is situated too far from the real source or too near to the real source node, it is not effective to preserve the source node's location privacy. Hence, in the given Fig. 6, *f2, f3, f4* are the best location for the fake source.

3.1. Phantom Routing

Both techniques *baseline flooding* and *single-path routing* are not much effective individually. The path from the source to the base station is fixed in both. As a result, the source node can be easily traced back by an adversary. However, the combination of these two protocols, called "Phantom routing," was a better choice. The outcomes of phantom

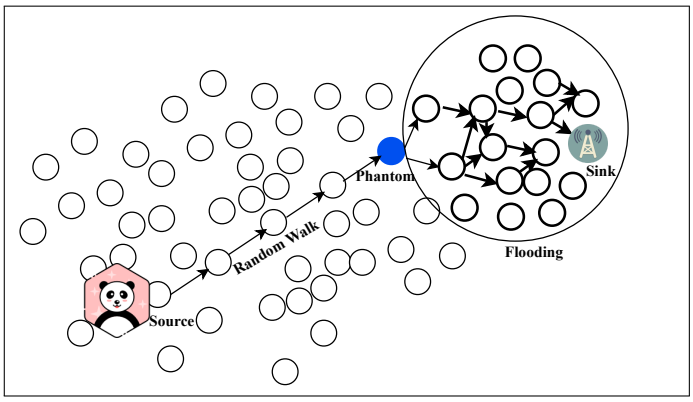


Fig. 5. Phantom routing for source location privacy

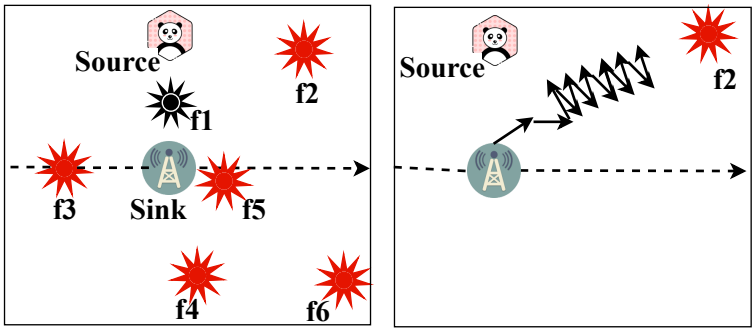


Fig. 6. Different location of fake source and message pulling direction

routing protocols confirmed that it is comparatively better in terms of safety than earlier proposed protocols.

In [23] authors focused on the weak points of the model developed by [17]. They said that if the location of the fake source node is fixed, then there is a high chance that an adversary will record the location and decrease the node's safety period. If the number of fake sources increases and the location of that fake source changes dynamically, it increases the source node's safety period. A more number of fake sources lead to more energy consumption, hence, will degrade network lifetime . To overcome this problem author proposed a protocol called *Cyclic Entrapment Method* (CEM). In CEM, between the source node and base station, a cycle is formed with fake messages as shown in Fig. 7. When an adversary starts backtracking from the base station node, they are trapped into the fake cycle. As a result, the source node's safety period is increased. The safety period depends on the number of loops activated between the source and base station nodes. If there are more loops, then the safety period is high.

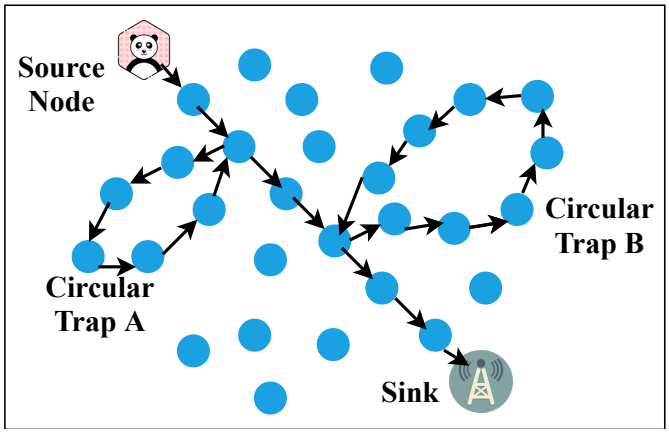


Fig. 7. Cyclic entrapment method

Shao et al. [24] proposed a model called *FitProbRate* that provides privacy to the source node from an adversary that can see the flow of the whole network at any time. In their model, each node forwarded a dummy packet in an interval. The interval may be fixed or probabilistic. Due to the probabilistic flooding technique, they claimed that it increased the source node's network lifetime and privacy. Wang et al. [25] mainly focused on the drawbacks of the phantom single path routing technique. They said that increasing the path length can't improve the safety period in the phantom single path routing technique—the proposed phantom routing with location angle (PRLA). PRLA works in two phases. First, using the inclination angle, selected the phantom node. Instead of choosing the fixed path for the random walk, the sensor node neighbours are divided into two different sets called *nearer* and *further* neighbour. As can be seen from Fig. 8, four phantom nodes are present, and all of them are in the range of the random walk. The message transmitting rate of the source node is to keep high compared to the message tracing rate to secure the source node location. Also, if the phantom node is just opposite the source

node, there was a very low probability of an adversary going there. Hence, the selection of such an area for the phantom node may be wasted, as shown in Fig. 9. The definition of a wasted path: The area of coverage that does not increase the privacy of the source node, called the wasted path. If the path having a minimum distance from the sink node to the phantom node crosses through the covered area, the transmitting period is more than the safety period.

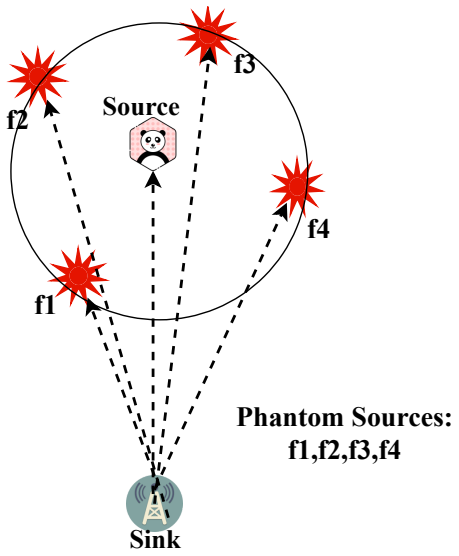


Fig. 8. Possible routes of the messages

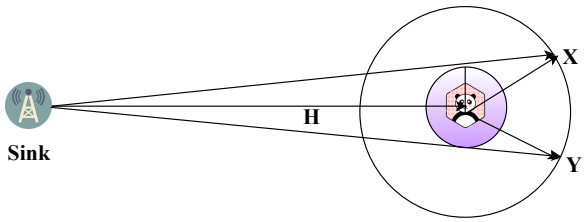


Fig. 9. The ratio of waste path

Doomun et al. [26] proposed a model called Source and Destination seclusion using Clouds (SECLOUD), which can hide the source node from the adversary. They used the concept of the fake source and fake base station. The source and base station nodes are hidden with the help cloud formed with a group of sensor nodes having similar configurations. The source node chose the cloud's size; if the size of the cloud was bigger, the

safety period of the source node was high and vice versa. Also, formed some fake source cloud and the base station cloud and were also similarly communicated with each other that the real source and real base station node is communicated. The source node's privacy increases if it intersects the communication line of the fake source cloud and real source cloud with each other at any point. Their model achieves privacy level comparatively better than the random walk technique regarding message overhead, anonymity, and unlike-ability. Recently a similar concept was used to protect the location of source [27].

Wang et al. [28] proposed a protocol called Weighted Random Stride (WRS) routing . In this technique, two parameters, the stride and the forwarding angle, were used (Fig. 10). The forwarding angle is between the estimated forwarding route and the line joining the forwarding node and the base station node. The main aim of their work was to fix up some pre-route from the source node to the base station node. The source node can choose any pre-set routes to send a message to the base station. So, an adversary is forced to stay on one of these routes to backtrace the source node's location, which helps to increase the safety period of the node. The application demand led to the development of data centre

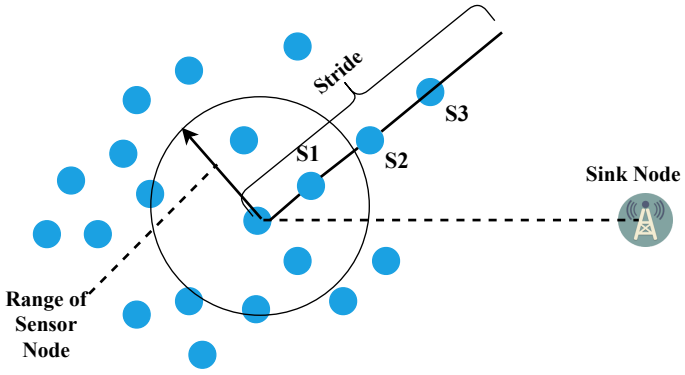


Fig. 10. Weighted random stride routing scheme

sensor networks. Shao et al. [29] proposed a protocol called Data-Centric Sensor networks (DCS) Security and Privacy Support. They named the sensor data based on event type or geographic location as a contrast to sensor nodes. To address DCS security issues, they proposed another protocol called a privacy-enhanced DCS (pDCS) network that offers different data privacy levels based on different cryptography keys. They also proposed query optimization techniques based on Euclidean Steiner Tree [30], and Keyed Bloom Filter [31] to minimize the query overhead while providing query privacy.

Alomair et al. [32] proposed a model that can guarantee the event indistinguishability by achieving Event Indistinguishability (EI) and interval indistinguishability (II). In EI, an adversary is unable to distinguish between the real event message and the fake message. In II, the adversary cannot distinguish between the first, the middle, or the end of the interval. The EI-based approach provides anonymity under EI and quantifies its information leakage. Their proposed technique was helpful to preserve the souce anonymity in the

wireless sensor networks. When a source node is located far away from the base station node, there is a high latency in the message delivery rate. Kokalj et al. [33] referred to this latency as the publishing route latency. They argue that the FitProbRate protocol [24] does not work well for the networks where the source node is just one hop away from the base station. The actual latency of the publishing route depends on the rate of reporting of events.

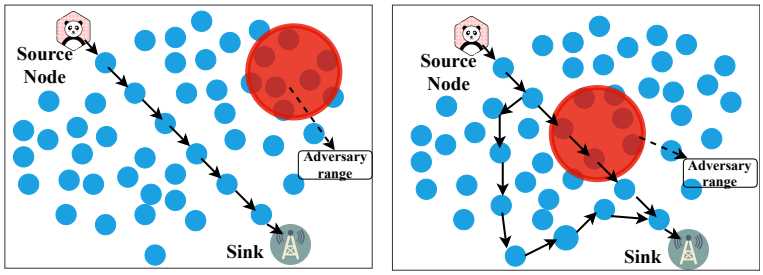


Fig. 11. Method to chose different path when an adversary is present

Rios et al. [34] proposed Context-Aware Location Privacy (CALP) protocol for SLP. Earlier proposed protocols have some disadvantages, such as in most of the techniques, the data packets were routed to the base station blindly without any prior knowledge of an adversary. Hence, it cost more energy consumption of the network that leads to network lifetime decreases. To overcome this issue, they used the advantages of sensor nodes, as the sensor node check whether any mobile agent is present in their communication range or not. Based on the adversary's availability, the process of data delivery to the base station is changed. The CALP algorithm's working principle can be seen from Fig. 11 where the network adapts the routing path to bypass an adversary moving in the locality of the shortest path.

Jiang et al. [14] described the importance of securing location information. The intruders can get the location information from the packets exchanged between the source and destination. They have done an extensive survey to classify the source location privacy, sink location privacy for securing the location information. Additionally, network performance, packet delay, energy efficiency, and network safety is analyzed. Mehta et al. [38] proposed a protocol that could hide the source location from the global adversary. A real sender can be hidden from an adversary by using multiple proxy source nodes [39]. The multiple source branch can be achieved by using the random walk model. The multiple proxy nodes are selected randomly from the list of neighbour nodes by the source node. This scheme prevents adversaries from getting the location information from the real source nodes. Besides, branch interference is created around the base station by increasing the routing branches.

Multiple sinks are used in this scheme to protect the source node from the adversaries [40]. Dynamically multiple paths are generated to confuse the adversaries. A high volume

Table 2. An overview of the research work for SLP in terms of different metrics

Protocol	**Accuracy**	**Power Uses**	**Delay**	**Privacy**
Fake source [6]	There is no impact on accuracy and data arrival	High	No	Misguide from real source
Dummy injection to protect real source [18]	There is no impact on accuracy and data arrival	High	Yes	Dummy packet disturb the traffic pattern
Flooding to protect the data source [36]	Baseline Flooding: Yes, Probabilistic Flooding:No Guarantee of data arrival	High	Baseline Flooding: NO, Probabilistic Flooding: Not guarantee packet comes with shortest route	Baseline Flooding: Less, Probabilistic Flooding: High
Random Delay [37]	There is no impact on accuracy and data arrival	Normal	Yes	Yes
Random Walk [5]	Phantom: Yes, Grow: Depends on intersection of random walk	Average	For Phantom: depends on no, of hops, GROW: depends on randomness path	Yes, Misguide from the real route
Random packet sending time [19]	There is no impact on accuracy and data arrival	Normal	Yes	Create ambiguity between two hops
Packet transmission rate	There is no impact on accuracy and data arrival	Normal	Yes	Hide traffic pattern with transmission control

of the number of packets can be transmitted through multiple paths towards multiple sink nodes. The scheme focuses on local adversaries based on the transmission loop of the actual and forgery packets. The sociality among the sensor nodes can be discontinued in this scheme.

Path Extension Method (PEM) proposed by Tan et al. [41] to preserve the privacy of the source node from the adversary. They used fake source concept to mislead the adversary from the real route. The fake source node is chosen and wherein the network they were placed as described in [6,28]. Fake sources are generated after the network is deployed and activated by receiving a message generated by the source node, increasing the sensor node lifetime. Once the fake sources receive a real message from the source node, they start creating a fake tree in the backbone with fake messages as shown in Fig. 12. The author compares their work with the other existing work based on fake source nodes such as [6,17], and found that the safety period of PEM is comparatively better than the existing work. Also, the delay is less, and the network lifetime is more.

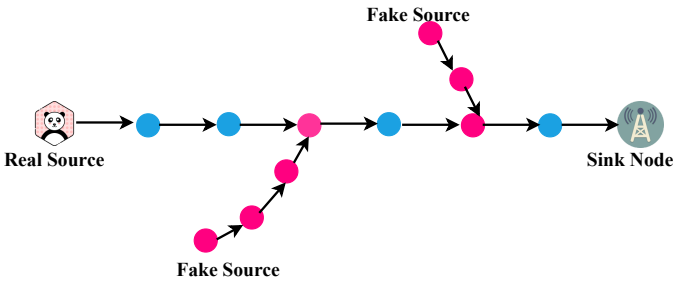


Fig. 12. The working model of path extension method

Angle-based Dynamic Routing Scheme (ADRS) proposed for SLP in [42]. When a sensor node becomes the source node, before broadcasting their message to all their neighbours' nodes, send a Request to Send (RTS) message. After getting the Clear to Send (CTS) from the neighbour sensor node, measure the angle ϕ between the neighbour node and the distance concerning the base station node. Based on these measurements, the next node was selected for the communication. A cloud-based source location privacy scheme is proposed with multiple sinks in place [43]. Due to the availability of multiple sinks, the destination of each packet changes randomly for each transaction. The routing paths are varied for each packet with the help of intermediate nodes. A directed random walk model is adopted to hide the source nodes' direction information from the adversaries. Roy et al. [5] used fake source and phantom routing concept for SLP. The source node sent the message to the phantom source using random walk. The phantom source flooded the message in the network to deliver it to the base station. Their model consumes more energy for flooding operations.

Zhou et al. [44] proposed an anonymous routing protocol for preserving location information (ARPLP) by using the proxy source nodes. The proxy node is randomly selected from the list of neighbours to create confusion for the hackers to get the location information of the actual source node. The real source node randomly sends the packets

to the neighbours until the packet reaches the proxy node. The routing branches are increased to disturb the adversary from getting the real path towards the source node. This scheme preserves the source/sink location information of the wireless multimedia sensor networks [45]. It uses multipath routing to hide the location information and the event occurrence of the source node. The cross-layer design among the application and routing layers is used to protect the source locations. Han et al. [46] proposed a model using the cloud and multi sink technology. The destination of every packet changes randomly for each transmission. The routing path for each packet changes automatically due to the presence of multiple sinks and intermediate nodes. Fake messages are added to the WSN to create a fake hotspot in the network. The important packets travel in multiple paths, which creates confusion for the hotspot locating adversaries.

Kumar et al. [19] proposed a new concept to secure the location of the source node in WSN. In their proposal, the base station node selected three nodes located on a fixed angle position that formed a triplet. Among these triplets, if a node becomes the source node, the other two-node act as a phantom node for that source node. Their work was extended by [36]. The authors used two phantom nodes to preserve the source node's privacy. The selection of the phantom node is based on the triplets. The triplets are a group of three nodes in the network based on a position concerning the sink node and the distance from the sink. Whenever the source node wants to send a packet to the base station, two phantom nodes are created and based on these. The packet is forwarded to the sink node via a phantom node. Since the phantom node's position and location are dynamic, it is very hard for an adversary to trace back the location of the source node.

Bai and Zhu [47] proposed an SLP scheme using a random annular region. The annular region was developed based on the coordinates of the actual and intermediate nodes. The relay node and source node are selected in one direction, but the phantom node was selected in the random annular region. The packets can be transmitted from the source towards the sink by strategically positioned mediate nodes [48]. The mediate nodes are selected based on the locality information. Multiple paths are used to transmit the packets towards the sink node, which creates confusion for the sink node to trace out the real sending node. An SLP scheme based on the anonymity cloud is proposed by Wang et al. [27]. The source node initially sends a lightweight message to its neighbours to create an anonymity cloud in its periphery. The set of nodes present in the cloud must have the same frequency range. The small message travel range forms the anonymity cloud. The duplicate nodes present in the borders of the cloud independently send the short messages. The real message can be recovered at the sink node when it receives at least t shares. The adversary uses the hidden Markov model to find the source node. So a probabilistic SLP model was proposed to identify the adversaries easily. The fake source nodes are used to mimic the behaviour of the actual source nodes. These specific nodes are used to diversify the routing path among the source and sink nodes. Deciding the next-hop node is dependent on the calculated weight of the nodes [49].

Bradbury et al. [50] proposed a hybrid model called DynamicSPR, which preserves the privacy of the source node. They used a random walk technique for fake source allocations in the network, which helped reduce energy consumption and improve the privacy of the source node. Wang et al. [51] proposed a model using the fake source technique, namely SLP full protection (SPFP). Their model able to defend the smart adversary also means the adversary who has access to both the global and local view of the network. To

address the issue of energy consumption in WSN, authors [52]proposed Energy Balanced Branch Tree (EBBT) in SLP. Their model uses fake sources and works in three phases: firstly, the source node is sent to an intermediate node randomly, then with minimum hop routing, the shortest path between the intermediate node and the base station is identified. Finally, a tree-shaped structure helps to achieve the privacy of the source node. Mamoun et al. [53] proposed clustering-based approach for SLP. They used dynamic shortest path and dynamic tree and their combinations to achieve the best privacy for the source. Chen et al. [54] suggested a protection scheme based on sector phantom routing scheme for SLP. Both the phantom source and random routing strategies were used in their approach. Tang et al. [55] suggested a theoretical model for analysing information leakage. Arvarsi et al. [56] did a survey of existing SLP protocols and calculated the number of sensor nodes needed to deploy to achieve the SLP, ensuring the connectivity of the WSN.

Alzaabi et al. [57] have presented a new location privacy protection algorithm based on phantom technique. This is energy-aware privacy preservation named phantom++. They have used a layering in-depth scheme to enhance the security in phantom++. The major problem with this scheme is validation because the authors have not presented any simulation results and analysis. To increase the source location privacy, fake packets and multi-path techniques are applied. An Adaptive Trust Sector-Based Authenticated System (ATSAS) is developed for SLP by Arivarasi, and Ramesh [58]. In their work, message authentication is done by honey encryption, and for security, packet encryption is used. This scheme provides better security in SLP, but it is complex due to multiple encryptions. Zhou et al. have designed a pseudospiral-based routing protocol for WSN to protect the node location as well as the location of base station [59]. To achieve this, they use a new two-phase location attack for two important types of nodes (including a base station and a source node). Mutalemwa and Shin extended their previous work to increase the reliability of the messages in SLP schemes [60]. They have done this work with three objectives. First, a new relay ring routing (ReRR) protocol is proposed, whereas in second, measuring the safety period of SLP with different parameters is done, and last, reliability of the scheme is evaluated.

George and Babu [61] proposed a semantic clustering-based approach to gain the efficacy of the source location privacy of the nodes. They encrypt each message sent by the sender to the intermediate nodes to minimize the chance of eavesdropping. The message transferred by each of the senders follows energy efficiency mechanisms to enhance the lifetime of the networks. The authors have assumed that the position of the base station is known to the attacker. So this tree-based clustering approach gives a better result as compared to the previous ones. The semantics co-relation-based clustering mechanism has shown better performance metrics such as energy consumption and message overhead in opposition to a universal adversary. The sink node verifies the identity of each source node by using a pseudo-random number. The authors tried to improve the data transmission process by combining the AES with ECC to minimize the chance of detecting the source location of the sender node [62]. A new localization method is proposed by the authors, which reduces the chance of localization error. The network communication overhead is also minimized by employing the authentication process for each sent information.

The summary of the research work published by the researcher using the fake sources and phantom routing techniques are presented in Table 3. To get the network information,

Table 3. A summary of key researches with network view and protocol used to preserve the SLP in WSNs

Proposed by	**Network View**	**Technique/Protocol**	**Issues**
Kamat el al. [6]	Local	PFSR,SLFSR	Yes
Majeed et al. [63]	Local	TARP	Yes
Roy el al. [5]	Local	FSAPR	NA
Kumar el al. [19]	Local	FSAPR	NA
Gupta el al. [36]	Local	2PARS	NA
Mahmoud et al. [4]	Local	CSPSLP	NA
Zhou et al. [39]	Local	Multiple proxy source nodes	NA
Hao et al. [49]	Local	Fake source nodes	Yes
Almalkawi et al. [45]	Local	WMSN and Multipath routing	NA
Han et al. [46]	Local	Cloud and Multisink	NA
Mutalemwa and Shin [22]	Local	Fake source routing	Yes
Bai et al. [47]	Local	Phantom node	Yes
Adilbekov et al. [21]	Local	Fake source node	NA
Wang et al. [20]	Local	PNCS	Yes
Zhou et al. [44]	Local	Proxy source node	NA
Mutalemwa et al. [48]	Local	Multiple path routing	Yes
Shao et al. [24]	Global	ProbRat and FitProbRate	Yes
Doomun et al. [26]	Global	SECLOUD	Yes
Yang el al. [64]	Global	PFS and TFS	Yes
Mehta et al. [38]	Global	PBA, SoSi	NA
Bicakci el al. [65]	Global	PBA	NA
Yang el al. [66]	Global	TCH-WSN	NA
Ortolani el al. [67]	Global	UHT	NA
Lu el al. [68]	Global	$TESP^2$	NA
Ouyang et al. [69]	Global	GOA	Yes
Kokalj el al. [33]	Global	GAFG	NA
Shao et al. [29]	Global	Used fake packets	NA
Yang et al. [70]	Global	ASLP	NA
Abbasi et al. [72]	Global	DRAA	Yes
Tangil et al. [73]	Global	DWUS	Yes
Jhumka et al. [18]	Global	FS1 & FS2	Yes
Han et al. [40]	Global	Multiple sinks and Fake packets	NA
Miao et al. [43]	Global	Multiple sinks	Yes
Chen et al. [71]	Both Local and Global	DBT & ZBT	NA

an adversary may perform a passive or active attacks. The form of attacks is explained as follows:

- **Denial of service:** This is an active attack in which an adversary is able to restrict all further communication between the nodes using the denial of service attacks.
- **Node Compromise:** In WSN, there is a high possibility that the nodes are getting compromised during the communication. There are two different types of node compromise, a) active node compromise and b)Passive node compromise.
- **Packet alteration:** It may be possible that an adversary altered the content of the packet before forwarding to the next hop.
- **Packet drops:** It may be possible that an adversary drops the incoming packet in between.
- **Packet injection:** The adversary is able to inject its own packet on the network.
- **Rate monitoring:** This is a passive attack which comes under traffic analysis attack. Through this attack an adversary looking for those sensor nodes which have a higher transmission rate. Such a node might be closer to either source or sink.
- **Angle of Arrival:** This is a passive attack that allows an adversary to see the incoming packet direction. An adversary needs a sectional antenna (special hardware) to perform this operation.
- **Hop-by-Hop trace:** An adversary able to follow the path of incoming message direction, using this they can easily reach the source of the message.
- **Eavesdropping:** An adversary is capable to overhear or intercept the message but can not decode them. They are only able to see the content of the message with this attack.
- **Timing analysis:** With the help of this attack, an adversary is able to understand the structure of the wireless sensor network.
- **Time correlation:** This is a passive attack in which timing information is used by an adversary to find out the path between source node to the sink node.
- **Traffic analysis:** The adversary has performed a traffic analysis of the WSN to analyze the path between the source and sink node. There is no specific method of traffic analysis is explained. It may be performed with the help of "rate monitoring", "timing analysis" attack.

Based on the capability, an adversary may view the entire network communication, or a part of network communication at a time. The different types of network access by the adversary are explained as follows.

- **Local view:** In this network view, an adversary is able to view only local (i.e., within their range) communication of the network.
- **Global view:** In this network view, an adversary is able to view the communication of the entire network [68,65,66].
- **Multi-Local view:** In this network view, there are many adversaries present in the networks and located at different network locations. Also, they are exchanging their information with each other. The other type of multi-local adversary includes a semi-global adversary, which is more powerful than the local adversary.

There may be a chance that the network nodes are compromised, and hence the important information may be shared with the adversary. For example:

- **Distribution of event:** An Adversary knows how the event is distributed in the network.
- **The protocol:** The adversary knows which protocol is used in the networks.
- **Identities of node:** An adversary knows the identity of the mentioned node in the message.
- **Location of the sink:** The adversary knows the location of the sink node.
- **Part of the routing algorithm:** The adversary knows some parts about the routing algorithm used on the network.

In the last column of Table 3 we have a parameter *Issue* which can be treated as follows: a) an adversary knows how the event is distributed in the network, b) adversary knows which protocol was used in the networks. Due to these issues, the source node's privacy may not be preserved for a longer time.

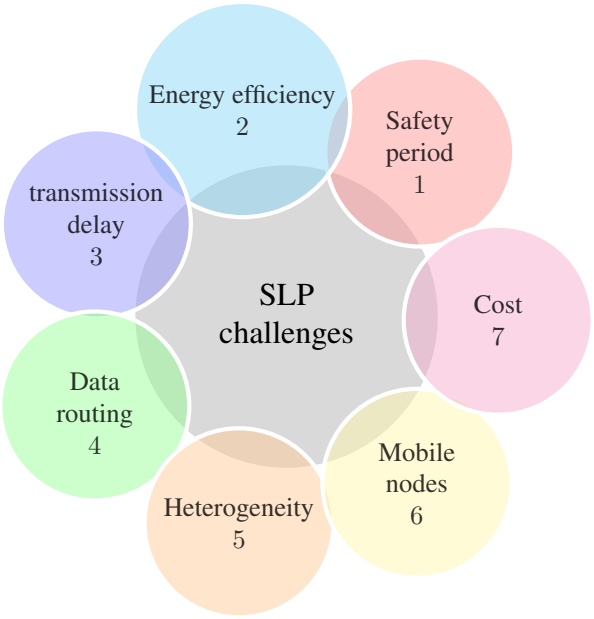


Fig. 13. SLP challenges of WSN with an increasing priority.

4. SLP challenges for WSN

The SLP model is an important area where many researchers have contributed their ideas and published papers. Our survey suggested several major challenges that required proper attention based on the reviewed journals in this area. These challenges are illustrated in Fig. 13 based on their priorities and discussed in detail. The priority is calculated depending on the published literature on these topics.

Safety period: The safety period is the total network duration earlier than the adversary is attacked or seized. It can be measured by calculating the total number of packets successfully transmitted to the destination [74]. The main challenge is how long the source node deviates from the adversary to reveal its location in this context. **Energy efficiency** Energy efficiency is always a vital parameter to increase the lifetime and performance of WSN. In SLP, we can save energy as well as privacy by using compressors or aggregators which aggregates the received data so that any adversary could not find out the original data and its source easily [75]. Privacy-preserving using aggregators to minimize the energy consumption are discussed in these articles [76,77,78].

Transmission delay: Maintaining transmission delay in the SLP context is the most challenging task. To preserve the source's location privacy, we need to deviate the routes of packets from source to destination or vice versa. It leads to extra delays in the network. Protocols proposed by [5] and [6] offer the best trade-off between transmission delay and SLP for WSN.

Data routing: To deliver data successfully from source to destination and vice versa, several routing schemes have been proposed like phantom routing, ring routing, random walk, etc. But still, more improvement is required. It is a major challenge to design an optimal routing scheme with a higher safety period to enhance the SLP [79].

Heterogeneity: Most of the research is done in homogeneous sensor networks for SLP, but real-world WSNs is heterogeneous. So we required some robust protocols for SLP, which can perform well in all scenarios. Design such types of schemes are a challenging task [80].

Mobile nodes: Nowadays, many WSNs are hybrid in terms of static and mobile nodes [81]. In mobile WSN, protecting the source is more challenging as compared to static [82]. Many researchers have proposed some SLP techniques using mobile nodes. **Cost:** Cost is always a major factor in any type of network. In WSN, to preserve the SLP, so many extra special nodes and devices are deployed, increasing the total cost of WSN. This can not be considered in most of the scenarios because one of the main advantages of WSN is the least cost [83].

4.1. Lesson Learned

Source location privacy in WSN is an important concern in the current era where everything is moving towards automation. The researchers propose many protocols to handle these issues using different approaches. The target of each approach is to preserve the location privacy of the source from the adversary. The popular technique used for this task is Phantom routing, where the source node delivers the message to the phantom node, and further, it is delivered to the base station using either the shortest path or flooding approach. Another popular approach to preserving the SLP is fake sources, which are created in the network. All created sources generate a similar message to the real one and flood the network. This message flood cerates challenge for the adversary to find the real one and trace the source node. Apart from this, the flooding technique is also used. In the flooding technique, the message is flooded in all directions of the network to deliver it to the base station. The researchers identified major issues with network lifetime. The sensors used in the network have limited power and can not be alive more time. Based on the network activities like sending the messages, receiving, or processing them, the power of sensor nodes decreases. Hence, to develop a robust privacy-preserving protocol,

energy consumption is an important parameter. Another parameter is latency. The aim of the network is to deliver sensitive information. Hence the latency must be minimum. The future researcher may considered these parameter for developing a new efficient privacy preserving protocol.

5. Future research directions

To secure the privacy of sources and data in WSN, new techniques with different domains increasing day by day. Researchers have tried many areas and techniques to improve SLP protocols' efficiency, but several areas are untouched or not explored properly. Fig. 14 illustrates some future research topics for SLP in WSN.

Content-oriented privacy: In the privacy model, several SLP schemes are designed as compared to content, or data-based privacy [84]. SLP is a part of content privacy only, but it requires special attention to keep its integrity, freshness, and confidentiality.

Energy harvesting: Energy harvesting in WSN is an emerging area of research. A few researchers have applied this energy harvesting to secure the source node [85]. However, this area has many possibilities to explore and develop some robust schemes for SLP.

Cloud-based: Cloud-based source location privacy is a new area where many scopes are there for new research. Few papers are published in this area where authors are used cloud with some fake sensor nodes to mislead the adversary [46] and [27].

Mobility: The impact of the mobility model on SLP is not being fully exploited in WSN. A mule mobility pattern [86] has been proposed to make a trade-off between SLP and delay. WE can also use mobile sinks or nodes to develop effective SLP techniques with minimum energy consumption.

Internet of Things (IoT): Now WSN is evolved with a new wing and applications known as IoT, which emerges new challenges and future works. The SLP is also important in IoT due to a large number of sensor nodes and real-time applications. IoT for SLP introduced [87], but a lot of things are required to do in this area.

Light-weight encryption algorithms: Several schemes have been proposed to protect the source from the adversary, but each has its problems. Most of the adversaries backtrack the source and destination messages and finds the actual source. A light-weight encryption algorithm is the most effective technique to protect the source but problem is designing a light-weight encryption algorithms [88][89]. In the future, designing a lightweight encryption protocol for WSN which suits the sensor nodes.

Network Coverage: Network coverage is an essential topic for all scenarios of the WSN. Many SLP related techniques are compromised with network coverage in the WSN [90]. To make a trade-off between SLP and network coverage is an important area where researchers need to give some more attention.

Hot Spot: Energy hot spot is a common phenomenon in WSN where a node consumes higher energy than other nodes, and that node dies early. Generally, the source and its nearby nodes transmit more packets than other nodes, resulting in an energy hot spot. Due to this problem, an adversary can easily detect the source, which is not acceptable in SLP methods [4]. So a lot of different things we can explore in SLP with an energy hot spot. In this area, a small number of researchers have shown their interest. Apart from the discussed future areas and challenges, many other sensor network-related issues come

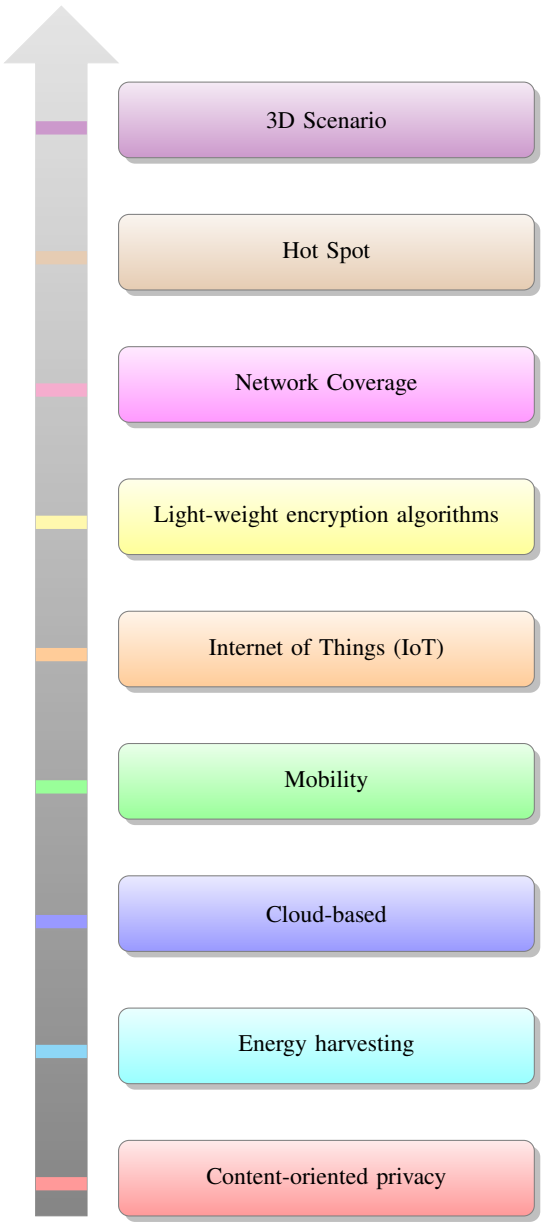


Fig. 14. Priority-wise research area for SLP in WSN

in the future, affecting SLP like new attacks or new technology. Researchers need to be ready for new solutions for new challenges and future works in WSN for SLP.

3D Scenario: Generally, authors consider a 2D scenario for their experiment in wireless sensor network but in the real-world 3D scenario is the best suited in this area [91]. Therefore, location privacy is also an important topic in 3D scenarios, but it has not received proper attention. In this area, a lot of potential work's scope is there, which the researchers need to deal with.

6. Conclusion

The source location privacy issue is continuously getting the attention of worldwide researchers, which shows the importance of this topic. Research has been done in this area, and many important milestones have been achieved, but certain issues still need to be explored. We have done an extensive survey on recently published papers on detecting SLP in Wireless Sensor Networks in this work. The Panda hunter game first explains the SLP problem. The WSN privacy issues are mainly categorized into two parts, a) Data privacy and b) context privacy. This article focused on the context of privacy, whose objective is to provide privacy to context, such as the location of the sensor nodes. The context of privacy is again divided into two categories: source location privacy and sink location privacy. In this paper, we intensely focused on source location privacy mechanisms.

The existing SLP schemes focused on local adversaries to be less capable of global and multi-local adversaries. Sending more fake packets attracts more energy consumption and maximum chance of congestion in the network. The SLP is not only possible to achieve by hiding the identity of a node. Additionally, dummy traffic, fake source nodes, and multi-path routing are used to counter the traffic analysis problem. In the future, this work can be extended by discussing the other location privacy issues such as sink location, temporal location privacy, and others. We discussed more than 90 papers on SLP, which were published in recent times. All the research works have been grouped based on their adversary model and network model. The readers of this paper will get insights into the different categories of SLP schemes used in WSNs. Lastly, the challenges of SLP towards the WSN is briefly discussed in this paper.

References

1. T. Qiu, R. Qiao, and D. O. Wu, "Eabs: An event-aware backpressure scheduling scheme for emergency internet of things," *IEEE Transactions on Mobile Computing*, vol. 17, no. 1, pp. 72–84, 2017.
2. P. Rawat, K. D. Singh, H. Chaouchi, and J. M. Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies," *The Journal of supercomputing*, vol. 68, no. 1, pp. 1–48, 2014.
3. B. Chakraborty, S. Verma, and K. P. Singh, "Differentially private location privacy preservation in wireless sensor networks," *Wireless Personal Communications*, vol. 104, no. 1, pp. 387–406, 2019.
4. M. M. Mahmoud and X. Shen, "A cloud-based scheme for protecting source-location privacy against hotspot-locating attack in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 10, pp. 1805–1818, 2012.

5. P. K. Roy, J. P. Singh, P. Kumar, and M. Singh, "3rd international conference on recent trends in computing 2015 (icrtc-2015) source location privacy using fake source and phantom routing (fsapr) technique in wireless sensor networks," *Procedia Computer Science*, vol. 57, pp. 936 – 941, 2015.
6. U. Kamat, Y. Zhang, and C. Ozturk, "Enhancing source-location privacy in sensor network routing," in *25th IEEE International Conference on Distributed Computing Systems (ICDCS*, 2005, pp. 599–608.
7. M. Conti, J. Willemsen, and B. Crispo, "Providing source location privacy in wireless sensor networks: A survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1238–1280, Third 2013.
8. N. Li, N. Zhang, S. K. Das, and B. Thuraisingham, "Privacy preservation in wireless sensor networks: A state-of-the-art survey," *Ad Hoc Networks*, vol. 7, no. 8, pp. 1501–1514, 2009.
9. J. Lopez and J. Zhou, "Wireless sensor network security, vol. 1 of cryptology and information security series," pp. 223–250, 2008.
10. S. Lee, J. Kim, and Y. Kim, "Preserving source-and sink-location privacy in sensor networks." *Comput. Sci. Inf. Syst.*, vol. 13, no. 1, pp. 115–130, 2016.
11. R. Rios, J. Lopez, and J. Cuellar, "Location privacy in wsns: solutions, challenges, and future trends," in *Foundations of Security Analysis and Design VII*. Springer, 2013, pp. 244–282.
12. J. Jiang, G. Han, H. Wang, and M. Guizani, "Privacy models in wireless sensor networks: A survey," *Journal of Sensors*, vol. 2016, pp. 1 – 18, 2016.
13. S. Gupta and B. Prince, "Preserving privacy of source location using random walk: A survey," in *2016 IEEE International Conference on Recent Trends in Electronics, Information Communication Technology (RTEICT)*, 2016, pp. 2047–2051.
14. J. Jiang, G. Han, H. Wang, and M. Guizani, "A survey on location privacy protection in wireless sensor networks," *Journal of Network and Computer Applications*, vol. 125, pp. 93–114, 2019.
15. M. Kamarei, A. Patooghy, A. Alsharif, and V. Hakami, "Simple: A unified single and multi-path routing algorithm for wireless sensor networks with source location privacy," *IEEE Access*, vol. 8, pp. 33 818–33 829, 2020.
16. V. Gomathy, N. Padhy, D. Samanta, M. Sivaram, V. Jain, and I. S. Amiri, "Malicious node detection using heterogeneous cluster based secure routing protocol (hcbs) in wireless adhoc sensor networks," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–7, 2020.
17. C. Ozturk and Y. Zhang, "Source-location privacy in energy-constrained sensor network routing," in *In ACM SASN*, 2004, pp. 88–93.
18. A. Jhumka, M. Leeke, and S. Shrestha, "On the use of fake sources for source location privacy: Trade-offs between energy and privacy," *The Computer Journal*, vol. 54, no. 6, pp. 860–874, 2011.
19. P. Kumar, J. P. Singh, P. Vishnoi, and M. P. Singh, "Source location privacy using multiple-phantom nodes in wsn," in *TENCON 2015 - 2015 IEEE Region 10 Conference*, Nov 2015, pp. 1–6.
20. Q. Wang, J. Zhan, X. Ouyang, and Y. Ren, "Sps and dps: Two new grid-based source location privacy protection schemes in wireless sensor networks," *Sensors*, vol. 19, no. 9, p. 2074, 2019.
21. U. Adilbekov, A. Adilova, and S. Saginbekov, "Providing location privacy using fake sources in wireless sensor networks," in *2018 IEEE 12th International Conference on Application of Information and Communication Technologies (AICT)*. IEEE, 2018, pp. 1–4.
22. L. C. Mutalemwa and S. Shin, "Secure routing protocols for source node privacy protection in multi-hop communication wireless networks," *Energies*, vol. 13, no. 2, pp. 1–30, 2020.
23. Y. Ouyang, Z. Le, G. Chen, J. Ford, and F. Makedon, "Entrapping adversaries for source protection in sensor networks," in *Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks*. IEEE Computer Society, 2006, pp. 23–34.
24. M. Shao, Y. Yang, S. Zhu, and G. Cao, "Towards statistically strong source anonymity for sensor networks," in *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*. IEEE, 2008, pp. 466–474.

25. W.-P. Wang, L. Chen, and J.-X. Wang, "A source-location privacy protocol in wsn based on locational angle," in *2008 IEEE International Conference on Communications*. IEEE, 2008, pp. 1630–1634.
26. R. Doomun, T. Hayajneh, P. Krishnamurthy, and D. Tipper, "Secloud: Source and destination seclusion using clouds for wireless ad hoc networks," in *Computers and Communications, 2009. ISCC 2009. IEEE Symposium*. IEEE, 2009, pp. 361–367.
27. N. Wang, J. Fu, J. Li, and B. Bhargava, "Source-location privacy protection based on anonymity cloud in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 15, no. 1, pp. 100–114, 2020.
28. H. Wang, B. Sheng, and Q. Li, "Privacy-aware routing in sensor networks," *Computer Networks*, vol. 53, no. 9, pp. 1512–1529, 2009.
29. M. Shao, S. Zhu, W. Zhang, G. Cao, and Y. Yang, "pdcs: Security and privacy support for data-centric sensor networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 8, pp. 1023–1038, 2009.
30. P. Winter and M. Zachariasen, "Euclidean steiner minimum trees: An improved exact algorithm," *Networks*, vol. 30, no. 3, pp. 149–166, 1997.
31. B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Communications of the ACM*, vol. 13, no. 7, pp. 422–426, 1970.
32. B. Alomair, A. Clark, J. Cuellar, and R. Poovendran, "Statistical framework for source anonymity in sensor networks," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. IEEE, 2010, pp. 1–6.
33. S. Kokalj-Filipović, F. Le Fessant, and P. Spasojević, "The quality of source location protection in globally attacked sensor networks," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*. IEEE, 2011, pp. 44–49.
34. R. Rios and J. Lopez, "Exploiting context-awareness to enhance source-location privacy in wireless sensor networks," *The Computer Journal*, pp. 1–11, 2011.
35. P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing source-location privacy in sensor network routing," in *Distributed Computing Systems, 2005. ICDCS 2005. Proceedings. 25th IEEE International Conference on*. IEEE, 2005, pp. 599–608.
36. S. Gupta, P. Kumar, J. P. Singh, and M. P. Singh, "Privacy preservation of source location using phantom nodes," in *Information Technology: New Generations*, S. Latifi, Ed. Cham: Springer International Publishing, 2016, pp. 247–256.
37. J. Chen, Z. Lin, Y. Liu, Y. Hu, and X. Du, "Sink location protection protocols based on packet sending rate adjustment," *International Journal of Distributed Sensor Networks*, vol. 12, no. 1, pp. 1–10, 2016.
38. K. Mehta, D. Liu, and M. Wright, "Protecting location privacy in sensor networks against a global eavesdropper," *IEEE Transactions on Mobile Computing*, vol. 11, no. 2, pp. 320–336, 2012.
39. L. Zhou and Y. Shan, "Multi-branch source location privacy protection scheme based on random walk in wsns," in *2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA)*. IEEE, 2019, pp. 543–547.
40. G. Han, H. Wang, X. Miao, L. Liu, J. Jiang, and Y. Peng, "A dynamic multipath scheme for protecting source-location privacy using multiple sinks in wsns intended for iiot," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 8, pp. 5527–5538, 2019.
41. W. Tan, K. Xu, and D. Wang, "An anti-tracking source-location privacy protection protocol in wsns based on path extension," *IEEE Internet of Things Journal*, vol. 1, no. 5, pp. 461–471, 2014.
42. P. Spachos, D. Toumpakaris, and D. Hatzinakos, "Angle-based dynamic routing scheme for source location privacy in wireless sensor networks," in *2014 IEEE 79th Vehicular Technology Conference (VTC Spring)*. IEEE, 2014, pp. 1–5.

43. X. Miao, G. Han, Y. He, H. Wang, and J. Jiang, "A protecting source-location privacy scheme for wireless sensor networks," in *2018 IEEE International Conference on Networking, Architecture and Storage (NAS)*. IEEE, 2018, pp. 1–5.
44. L. Zhou, Y. Shan, and X. Chen, "An anonymous routing scheme for preserving location privacy in wireless sensor networks," in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*. IEEE, 2019, pp. 262–265.
45. I. T. Almalkawi, J. Raed, N. Alghaeb, and M. G. Zapata, "An efficient location privacy scheme for wireless multimedia sensor networks," in *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2019, pp. 1615–1618.
46. G. Han, X. Miao, H. Wang, M. Guizani, and W. Zhang, "Cpslp: A cloud-based scheme for protecting source location privacy in wireless sensor networks using multi-sinks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2739–2750, 2019.
47. L. Bai, H. Zhu, and G. Li, "Privacy protection algorithm based on random annular region in wsn," in *2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS)*. IEEE, 2018, pp. 64–67.
48. L. C. Mutalemwa and S. Shin, "Strategic location-based random routing for source location privacy in wireless sensor networks," *Sensors*, vol. 18, no. 7, p. 2291, 2018.
49. H. Wang, G. Han, W. Zhang, M. Guizani, and S. Chan, "A probabilistic source location privacy protection scheme in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 5917–5927, 2019.
50. M. Bradbury, A. Jhumka, and M. Leeke, "Hybrid online protocols for source location privacy in wireless sensor networks," *Journal of Parallel and Distributed Computing*, vol. 115, pp. 67–81, 2018.
51. N. Wang, J. Fu, J. Zeng, and B. K. Bhargava, "Source-location privacy full protection in wireless sensor networks," *Information Sciences*, vol. 444, pp. 105–121, 2018.
52. H. Wang, L. Wu, Q. Zhao, Y. Wei, and H. Jiang, "Energy balanced source location privacy scheme using multibranch path in wsns for iot," *Wireless Communications and Mobile Computing*, vol. 2021, 2021.
53. M. F. Al-Mistarihi, I. M. Tanash, F. S. Yaseen, and K. A. Darabkh, "Protecting source location privacy in a clustered wireless sensor networks against local eavesdroppers," *Mobile Networks and Applications*, vol. 25, no. 1, pp. 42–54, 2020.
54. Y. Chen, J. Sun, Y. Yang, T. Li, X. Niu, and H. Zhou, "Psspr: A source location privacy protection scheme based on sector phantom routing in wsns," *International Journal of Intelligent Systems*, 2021.
55. D. Tang, J. Gu, W. Han, and X. Ma, "Quantitative analysis on source-location privacy for wireless sensor networks," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2020, pp. 805–809.
56. A. Arivarasi and P. Ramesh, "Review of source location security protection using trust authentication schema," in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*. IEEE, 2020, pp. 215–222.
57. A. Alzaabi, A. Aldoobi, D. Alnuaimi, L. Alserkal, M. Alsuwaidi, and N. Ababneh, "Grid-based source location privacy protection schemes in iot wireless sensor networks," in *2021 4th International Conference on Data Storage and Data Engineering*, 2021, pp. 31–36.
58. A. Arivarasi and P. Ramesh, "An improved source location privacy protection using adaptive trust sector-based authentication with honey encryption algorithm in wsn," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–13, 2021.
59. Z. Zhou, Y. Wang, P. Li, X. Chang, and J. Luo, "Node location privacy protection in unattended wireless sensor networks," *Mathematical Problems in Engineering*, vol. 2021, 2021.
60. L. C. Mutalemwa and S. Shin, "Novel approaches to realize the reliability of location privacy protocols in monitoring wireless networks," *IEEE Access*, vol. 9, pp. 104 820–104 836, 2021.

61. C. M. George and S. L. Babu, "A scalable correlation clustering strategy in location privacy for wireless sensor networks against a universal adversary," in *2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*. IEEE, 2019, pp. 1–3.
62. M. A. Tamtalini, A. E. B. El Alaoui, and A. El Fergougui, "Eslc-wsn: A novel energy efficient security aware localization and clustering in wireless sensor networks," in *2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*. IEEE, 2020, pp. 1–6.
63. A. Majeed, K. Liu, and N. Abu-Ghazaleh, "Tarp: Timing analysis resilient protocol for wireless sensor networks," in *2009 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*. IEEE, 2009, pp. 85–90.
64. Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao, "Towards event source unobservability with minimum network traffic in sensor networks," in *Proceedings of the first ACM conference on Wireless network security*. ACM, 2008, pp. 77–88.
65. K. Bicakci, H. Gultekin, B. Tavli, and I. E. Bagci, "Maximizing lifetime of event-unobservable wireless sensor networks," *Computer Standards & Interfaces*, vol. 33, no. 4, pp. 401–410, 2011.
66. Y. Yang, J. Zhou, R. H. Deng, and F. Bao, "Better security enforcement in trusted computing enabled heterogeneous wireless sensor networks," *Security and Communication Networks*, vol. 4, no. 1, pp. 11–22, 2011.
67. S. Ortolani, M. Conti, B. Crispo, and R. D. Pietro, "Events privacy in wsns: A new model and its application," in *World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2011 IEEE International Symposium on a*. IEEE, 2011, pp. 1–9.
68. R. Lu, X. Lin, H. Zhu, and X. Shen, "Tesp2: Timed efficient source privacy preservation scheme for wireless sensor networks," in *Communications (ICC), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1–6.
69. Y. Ouyang, Z. Le, D. Liu, J. Ford, and F. Makedon, "Source location privacy against laptop-class attacks in sensor networks," in *Proceedings of the 4th international conference on Security and privacy in communication netowrks*. ACM, 2008, pp. 1–10.
70. W. Yang and W. T. Zhu, "Protecting source location privacy in wireless sensor networks with data aggregation," in *International Conference on Ubiquitous Intelligence and Computing*. Springer, 2010, pp. 252–266.
71. H. Chen and W. Lou, "From nowhere to somewhere: protecting end-to-end location privacy in wireless sensor networks," in *International Performance Computing and Communications Conference*. IEEE, 2010, pp. 1–8.
72. A. Abbasi, A. Khonsari, and M. S. Talebi, "Source location anonymity for sensor networks," in *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*. IEEE, 2009, pp. 1–5.
73. G. Suarez-Tangil, E. Palomar, B. Ramos, and A. Ribagorda, "An experimental comparison of source location privacy methods for power optimization in wsns," in *Proceedings of the 3rd WSEAS international conference on Advances in sensors, signals and materials*, 2010, pp. 79–84.
74. X. Deng, X. Xin, and T. Gao, "A location privacy protection scheme based on random encryption period for vsns," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 3, pp. 1351–1359, 2020.
75. M. Alrashidi, N. Nasri, S. Khediri, and A. Kachouri, "Energy-efficiency clustering and data collection for wireless sensor networks in industry 4.0," *Journal of Ambient Intelligence and Humanized Computing*, 2020.
76. W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. Abdelzaher, "Pda: Privacy-preserving data aggregation in wireless sensor networks," in *IEEE INFOCOM 2007 - 26th IEEE International Conference on Computer Communications*, 2007, pp. 2045–2053.
77. Y. Yang, X. Wang, S. Zhu, and G. Cao, "Sdap: A secure hop-by-hop data aggregation protocol for sensor networks," vol. 11, no. 4, pp. 1–43, 2008.

78. Abizar, Farman, Jan, Khan, and Koubaa, "A smart energy-based source location privacy preservation model for internet of things-based vehicular ad hoc networks," *Transactions on Emerging Telecommunications Technologies*, pp. 1–14, 2020.
79. J. Wu, Z. Chen, and M. Zhao, "An efficient data packet iteration and transmission algorithm in opportunistic social networks," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–13, 2019.
80. A. S. H. Abdul-Qawy and T. Srinivasulu, "Sees: a scalable and energy-efficient scheme for green iot-based heterogeneous wireless nodes," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 4, pp. 1571–1596, 2019.
81. S. K. Singh and P. Kumar, "A load balancing virtual level routing (lbvlr) using mobile mule for large sensor networks," *The Journal of Supercomputing*, vol. 75, no. 11, pp. 7426–7459, 2019.
82. A. Aranganathan and C. Suriyakala, "An efficient secure detection and prevention of malevolent nodes with lightweight surprise check scheme using trusted mobile agents in mobile ad-hoc networks," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 9, pp. 3493–3503, 2019.
83. J.-A. Kim, D. G. Park, and J. Jeong, "Design and performance evaluation of cost-effective function-distributed mobility management scheme for software-defined smart factory networking," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–17, 2019.
84. G. DaSilva, V. Loud, A. Salazar, J. Soto, and A. Elleithy, "Context-oriented privacy protection in wireless sensor networks," in *2019 IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, 2019, pp. 1–4.
85. C. Huang, M. Ma, Y. Liu, and A. Liu, "Preserving source location privacy for energy harvesting wsns," *Sensors*, vol. 17, no. 4, p. 724, 2017.
86. M. Raj, N. Li, D. Liu, M. Wright, and S. K. Das, "Using data mules to preserve source location privacy in wireless sensor networks," *Pervasive and Mobile Computing*, vol. 11, pp. 244 – 260, 2014.
87. G. Han, H. Wang, J. Jiang, W. Zhang, and S. Chan, "Caslp: A confused arc-based source location privacy protection scheme in wsns for iot," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 42–47, 2018.
88. R.-h. Hu, X.-m. Dong, and D.-l. Wang, "Protecting data source location privacy in wireless sensor networks against a global eavesdropper," *International Journal of Distributed Sensor Networks*, vol. 10, no. 8, pp. 1–17, 2014.
89. L. Kazatzopoulos, C. Delakouridis, G. F. Marias, and P. Georgiadis, "ihide: hiding sources of information in wsns," in *Second International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing (SecPerU'06)*, 2006, pp. 1–8.
90. A.-S. Abuzneid, T. Sobh, and M. Faezipour, "An enhanced communication protocol for location privacy in wsn," *International Journal of Distributed Sensor Networks*, vol. 11, no. 4, pp. 1–15, 2015.
91. J. Kumari, P. Kumar, and S. K. Singh, "Localization in three-dimensional wireless sensor networks: a survey," *The Journal of Supercomputing*, vol. 75, no. 8, pp. 5040–5083, 2019.

Pradeep Kumar Roy received the B. Tech degree in Computer Science and Engineering from BPUT University Odisha. He received his M. Tech and Ph.D. degree in Computer Science and Engineering from the National Institute of Technology Patna in 2015 and 2018, respectively. He received a Certificate of Excellence for securing a top rank in the M. Tech course. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Indian Institute of Information Technology (IIIT) Surat, Gujarat, India. He also worked in Vellore Institute of Technology, Vellore, Tamil Nadu, India.

His area of specialization straddles across question answering, text mining and information retrieval, social network, and wireless sensor networks. He is part of the technical program committee and chaired many technical sessions of International Conferences. He has published articles in different journals, including IEEE Transaction on Artificial Intelligence, Neural Processing Letters, IJIM, Neural Computing and Applications, Future Generation Computer Systems, and others. He has also published the conference proceedings in various international conferences.

Asis Kumar Tripathy (SMIEEE, MACM, and MIE) is an Associate Professor in the School of Information Technology and Engineering, Vellore Institute of Technology, Vellore, India. He has more than ten years of teaching experience. He completed his Ph.D. from the National Institute of Technology, Rourkela, India, in 2016. His areas of research interests include wireless sensor networks, cloud computing, the Internet of things, and advanced network technologies. He has several publications in refereed journals, reputed conferences, and book chapters to his credit. Dr. Tripathy is serving as the associate editor of International Journal of Computational Science and Engineering (Inderscience). He has served as a program committee member in several conferences of repute. He has also been involved in many professional and editorial activities.

Sunil Kumar Singh is currently working as an Assistant Professor in the School of Computer Science and Engineering at VIT-AP University, Vijayawada, India. He has done his Ph.D. in Computer Science and Engineering form National Institute of Technology Patna, India in 2018. He received the M. Tech and B. Tech degrees in Computer science and Engineering and Information Technology, both from Kalyani Government Engineering College, Kalyani, India in 2010 and 2007, respectively. He has over 30 publications in various National/International Journals & Conferences (viz. IEEE, ACM, Springer and Elsevier). He is also the reviewer of several reputed journals indexed in SCI, SCIE and Scopus. He is also in the Program Committee of various National/International Conferences. He has delivered expert talks and guest lectures at various prestigious institutes. His research area includes Wireless Sensor Networks.

Kuan-Ching Li is currently appointed as Distinguished Professor at Providence University, Taiwan. He is a recipient of awards and funding support from several agencies and high-tech companies, as also received distinguished chair professorships from universities in several countries. He has been actively involved in many major conferences and workshops in program/general/steering conference chairman positions and as a program committee member, and has organized numerous conferences related to high-performance computing and computational science and engineering. Professor Li is the Editor-in-Chief of technical publications Connection Science (Taylor & Francis), International Journal of Computational Science and Engineering (Inderscience) and International Journal of Embedded Systems (Inderscience), and serves as associate editor, editorial board member and guest editor for several leading journals. Besides publication of journal and conference papers, he is the co-author/co-editor of several technical professional books published by CRC Press, Springer, McGraw-Hill, and IGI Global. His topics of interest include parallel and distributed computing, Big Data, and emerging technologies. He is a Member of the AAAS, a Senior Member of the IEEE, and a Fellow of the IET.

Received: October 09, 2021; Accepted: February 01, 2022.

Computer Science and Information Systems 19(2):887–912 https://doi.org/10.2298/CSIS211020015N

RG-SKY: A Fuzzy Group Skyline Relaxation for Combinatorial Decision Making

Sana Nadouri[1,2], Allel Hadjali[1] and Zaidi Sahnoun[2]

[1] LIAS/ISAE-ENSMA
Poitiers, France
sana.nadouri@{univ-constantine2.dz, ensma.fr}
allel.hadjali@ensma.fr
[2] LIRE/University of Constantine 2
Constantine, Algeria
zaidi.sahnoun@univ-constantine2.dz

Abstract. Skyline queries were recently expanded to group decision making to meet complex real-life needs encountered in many modern application domains that does not only require analyzing individual points but also groups of points. Group skyline aims at retrieving groups that are not dominated by any other group of the same size in the sense of a group-dominance relationship. It may often happens that this kind of dominance leads to only a small number of non-dominated groups which could be insufficient for the decision maker. In this paper, we propose to extend group skyline dominance by making it more demanding so that several groups leave incomparable. Then, the original group skyline will be enlarged by some interesting groups that are not much dominated by any other group. The key element of this relaxation is a particular fuzzy preference relation, named "much preferred", conveniently chosen. Furthermore, algorithms to compute the relaxed group skyline are proposed. Finally, a set of experiments are conducted on real, synthetic and generated data. Such experiments show that our proposal can really improve the decision process and satisfy user queries, insure reliability and decision quality.

Keywords: Data analysis, Group skyline queries, Relaxation, Fuzzy preferences, Decision making.

1. Introduction

Nowadays, multi-criteria analysis and decision making become more and more complex due to conflicting criteria and query complexity. Skyline operator [3], known as Maxima in computational geometry or Pareto in the business management field, manages this complexity using Pareto dominance. It extracts interesting objects from a dataset by respecting user preferences. It is particularly very successful in the database field, it has undergone an exponential interest due to the benefit that can be derived from it, even in real contexts. Skyline queries return then the most interesting points based on Pareto dominance relationship defined as follows: let a and b be two points (or database tuples) with the same number of attributes (also called dimensions), a dominates b, noted: ($a \prec b$), if a is as good as b in all dimensions and better than b in at least one dimension. If neither $a \prec b$ and nor $b \prec a$, then a and b are incomparable.

Many propositions and research works were published to study the skyline semantics and optimize its computation. Efficient algorithms were proposed to retrieve objects that present the optimal combination of the dataset characteristics [3,5,10,16,24,26,30,31]. Recently, the skyline definition turns out to be poor to deal with new complex real-world decision applications to answer different queries that require choosing a group of objects rather than individual objects of a dataset. Let us consider an example where a user wants to get the best Volleyball teams from a set of players, the traditional skyline returns the best players but not the best combinations to create a team that can not be dominated by any other existing team generated from the set of players. Another similar example consists of choosing a group of experts to review and evaluate papers based on the experts collective strength on multiple desired skills. This leads to the concept of Group Skyline [3] [14,38], noted G-SKY, which is very important and useful in many other domains like: groups recommendation, investments selection, detection of fire/crime (most dangerous places), etc. This novel concept has created new issues to the skyline community, for instance, generating groups and returning the appropriate skyline groups became the most challenging problems. Some solutions to these issues have been proposed in the literature [32,33]. However, querying a d-dimensional dataset using group skyline queries may lead to two particular scenarios: (i) a large number of skyline groups returned, which could be less informative for decision makers, (ii) a small number of skyline groups returned, which could be insufficient for decision makers. To solve the first problem (i), various approaches [15,29,40,43] are proposed to refine the group skyline, therefore reducing its size, but none of the existing work has addressed the problem (ii) to relax the group skyline in order to increase the number of group skyline results and thus satisfy better the decision makers needs.

Consider the following problem of finding the skyline l-groups (where l indicates the number of elements in each group from an n-tuple dataset): a decision maker (the trainer) wants to get the 5 best groups of 3 players by maximizing points and the number of blocks, when we run his/her query the system returns 2 groups of 3 Volleyball players each. Unfortunately, this answer does not satisfy the decision maker, he/she needs 5 teams of the best Beach-volleyball players but the traditional group skyline definition can return only 2 groups. It is the principal issue addressed in this paper. The solution advocated aims at enlarging the size of the group skyline by applying an appropriate relaxation process. This process consists of retrieving non-skyline groups by making more demanding the group-dominance relationship. To the best of our knowledge this is the first time the problem of group skyline relaxation is addressed.

Taking as starting point the study about the traditional skyline relaxation discussed in [2], we propose an extended group dominance relationship using a particular fuzzy preference relation, called *Much Preferred* (MP). The proposed approach allows increasing the group skyline with (non-skyline) groups that are only dominated to some extent by other groups in the sense of the extended group dominance introduced. The nature of the relation MP makes it more demanding the dominance between groups of the target dataset. In this context, a group still belong to the group skyline unless it is much dominated, in the spirit of the *MP relation*, by another skyline group. By this way, many groups are considered as incomparable and then as elements of the new relaxed group skyline (noted RG-SKY). Note that using the traditional group skyline definition such groups are pruned

[3] Named also combinatorial or compositional skyline

from the skyline groups. Furthermore, two algorithms (naive and optimized versions) with different cases to compute RG-SKY efficiently are provided. We also develop a set of experiments on real, synthetic and generated data to study and analyze the relevance and effectiveness of our proposal.

The remainder of the paper is organized as follows. In Section 2, we provide a necessary background and the problem description. In Section 3, we present a comparative study relying on a literature survey of related work. In Section 4, we define the RG-SKY concept and provide its properties w.r.t. both semantics and behavior. Algorithms for RG-SKY computation are also discussed. In Section 6, we evaluate the approach with different cases and discuss the results. Finally, Section 7 concludes by discussing the implications of this work in optimizing decision-making by increasing user satisfaction.

2. Background and Problem Description

This section presents a brief overview about the traditional skyline, group skyline and notions of fuzzy set theory used in this work. Then, it provides a description of the problem of interest. Table 2 provides the symbols with their meanings used in the rest of the paper.

Table 1. Symbols and their meanings

Symbol	Meaning
$\mathbb{D}$=(D_1, D_2, . . . , D_d)	*a set of d-dimensional data points*
d	*The number of dimensions of the set* $\mathbb{D}$
A_i	*The attribute* A_i
D_i	*The domain of* A_i
Q	*A point of* $\mathbb{D}$
g_i = (Q_1^i, ..., Q_l^i)	*A group of* l *points of* $\mathbb{D}$
$\mathbb{G}$ = (g_1, g_2, ..., g_l)	*The set of groups of size* l *of* $\mathbb{D}$
F	*An aggregation function*
MP	*Much preferred relation*
SKY	*The set of traditional skyline points*
G-SKY	*The group skyline (set of skyline groups)*
Rest	*The set of non skyline groups*
RG-SKY	*The relaxed group skyline*

2.1. Background

Traditional Skyline Queries Let $\mathbb{D}$ = (D_1, D_2, . . . , D_d) be a set of d-dimensional data points (that corresponds to a set of database tuples). We define a relation $R(A_1, A_2, ..., A_d)$ in $\mathbb{D}$ and we assume the existence of a total order relation on each domain D_i.

The traditional skyline query is based on Pareto dominance relationship defined as follows. Let a and b be two different points in $\mathbb{D}$, a dominates (in Pareto sense) b, denoted by a $\prec$ b, if for all i, a[i] $\leq$ b[i], and for at least one i, a[i] $<$ b[i], where a[i] (Resp. b[i]) is the value of the point a (Resp. b) for the attribute A_i and 1 $\leq$ i $\leq$ d. Formally,

$$a \prec b \Leftrightarrow \forall i \in \{1, ..., d\} : a[i] \leq b[i] \; and \; \exists j \in \{1, ..., d\} : a[j] < b[j] \qquad (1)$$

Without loss of generality, we consider in this definition the smallest value, the better.
The skyline of $\mathbb{D}$, denoted SKY($\mathbb{D}$), is a set of points that are not dominated (in Pareto

sense) by any other point from $\mathbb{D}$. Formally,

Hotels	Distance	Price	Is a Skyline ?
h1	1200	200	✗
h2	500	1400	✓
h3	800	2000	✗
h4	800	600	✓
h5	600	1400	✗
h6	100	2100	✓
h7	1000	100	✓

✓ Yes ✗ No

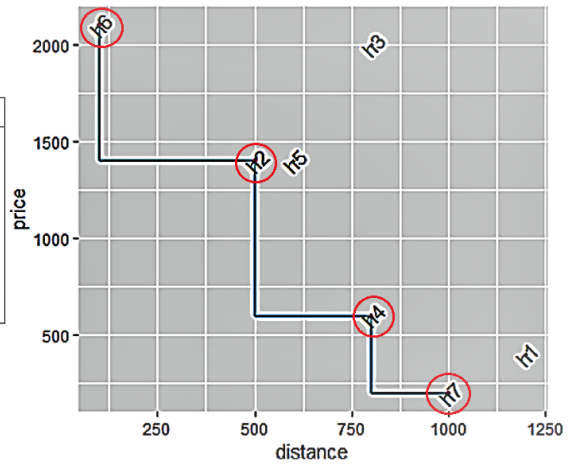


Fig. 1. The hotels conference Skyline points

The SQL[4] skyline query format is an SQL extension that incorporates a new clause **SKYLINE** where user preferences are specified [11]. The proposed SQL skyline syntax based on Borzsony [3] writes as follows:
"SELECT * FROM ... WHERE ... SKYLINE ... ORDER BY ..."

Example 1. Consider the example of a conference PhD student participant who wants to reserve a hotel room (see in Figure 1 the set of hotels). She/He wants the hotel to be close to the conference place and also to pay a reasonable price. The corresponding SQL skyline query writes:
"**SELECT** Hotels, Distance, Price **FROM** Hotels **SKYLINE** OF Price MIN, Distance MIN"
where MIN specifies that the two attributes should be minimized. Figure 1 shows the skyline points circled in red (i.e., $SKY = \{h_6, h_2, h_4, h_7\}$) that answer the user's query to get the best hotels that satisfy the students criteria. ■

Group Skyline Queries Despite the success of the traditional skyline, which focuses on top-1 solutions, it is inadequate when optimal groups are searched rather than individual points. For this reason, the skyline community proposed to extend the skyline definition to the combinatorial context to deal with group skyline instead of individual Skyline.

This new type of skyline queries extension relies on a dominance relationship between groups, also called combinatorial skyline queries. Group skyline returns groups that are not dominated by any other group in the Dataset. The dominance relationship between groups can be formulated in two different ways as follows:

– The first formulation relies on Pareto dominance and an aggregate function conveniently chosen. The dominance between groups is named *G-Skyline*.

[4] SQL stands for Structured Query Language· SQL allows us to access and manipulate databases

– The second semantics introduces a particular generalized dominance relation applied on permutations of groups to be compared. This type of dominance is named *G-Dominance*.

This leads thus to the two following formal definitions [43] (where g and g' are two groups with the same size):

Definition 1. ***(G-Skyline)****. Let F be an aggregate function and g = $\{Q_1, Q_2, ..., Q_l\}$ (resp. g' = $\{Q'_1, Q'_2, ..., Q'_l\}$) be a group represented by a point Q (resp. Q') with Q = F($Q_1, Q_2, ..., Q_l$) (resp. Q' = F($Q'_1, Q'_2, ..., Q'_l$)). For two distinct groups g and g', g dominates g' (denoted g $\prec_{gs}$ g') if Q dominates Q' (Q $\prec$ Q') in Pareto sense.*

Definition 2. ***(G-Dominance)****. Let us consider the two previous groups g and g', g G-dominates g' (denoted g $\prec_{gd}$ g'), if two permutations of l points can be found for g and g', g = $\{Q_{u1}, Q_{u2}, ..., Q_{ul}\}$ and g' = $\{Q'_{v1}, Q'_{v2}, ..., Q'_{vl}\}$, such that $Q_{ui} \preceq Q'_{vi}$ [5] for all i (1 $\leq$ i $\leq$ l) and $Q_{ui} \prec Q'_{vi}$ for at least one i.*

The group skyline of a dataset $\mathbb{D}$, denoted G-Sky($\mathbb{D}$), is a set of groups that are not dominated by any other group of $\mathbb{D}$ in the sense of definition 1 or 2. Formally, we write:

$$G - SKY(\mathbb{D}) = \{g \in \mathbb{G} \mid \nexists g' \in \mathbb{G} : g' \ltimes g\} \tag{3}$$

where $\ltimes$ stands for the relation $\prec_{gs}$ or $\prec_{gd}$.

Table 2. Players data

Players	points	rebounds
p_1	3	3
p_2	0	4
p_3	4	1
p_4	2	3
p_5	2	2
p_6	2	1

Example 2. Let us consider an example consisting of team players selection. Assume that we have six players p_1,...,p_6 shown in table 2 and we need team players formed by two players. In table 3, we generate all the possible groups of two players. The manager likes to extract the best teams based on the scored points and rebounds attributes of the players (i.e., **the greatest value, the better**).

– Based on the **Definition 1** and MAX aggregation function on each attribute, the values w.r.t. points and rebounds of each player of the generated teams are given in Column 3 of table 3. For instance, g_1 has two players p_1 and p_2, the values of $g_1 = (max(3, 0), max(3, 4)) = (3, 4)$. Now, applying the Pareto dominance on the generated teams, one can check that the group skyline contains only the team g_6.

[5] $Q_j \preceq Q'_j$ means that $Q_j \prec Q'_j$ or $Q_j \sim Q'_j$ where $\sim$ stands for the indifference relation (i.e., equally preferable). The indifference relation reduces to equality if each domain D_i is endowed with a total order.

Table 3. Skyline Teams (using definition 1)

Groups	Players	MAX (points,rebounds)	Group skyline
g_1	p_1(3,3) - p_2(0,4)	G_1(3,4)	✗
g_2	p_1(3,3) - p_3(4,1)	G_2(4,3)	✗
g_3	p_1(3,3) - p_4(2,3)	G_3(3,3)	✗
g_4	p_1(3,3) - p_5(2,2)	G_4(3,3)	✗
g_5	p_1(3,3) - p_6(2,1)	G_5(3,3)	✗
g_6	p_2(0,4) - p_3(4,1)	G_6(4,4)	✓
g_7	p_2(0,4) - p_4(2,3)	G_7(2,4)	✗
g_8	p_2(0,4) - p_5(2,2)	G_8(2,4)	✗
g_9	p_2(0,4) - p_6(2,1)	G_9(2,4)	✗
g_{10}	p_3(4,1) - p_4(2,3)	G_{10}(4,3)	✗
g_{11}	p_3(4,1) - p_5(2,2)	G_{11}(4,2)	✗
g_{12}	p_3(4,1) - p_6(2,1)	G_{12}(4,1)	✗
g_{13}	p_4(2,3) - p_5(2,2)	G_{13}(2,3)	✗
g_{14}	p_4(2,3) - p_6(2,1)	G_{14}(2,3)	✗
g_{15}	p_5(2,2) - p_6(2,1)	G_{15}(2,2)	✗

✓ Yes ✗ No

– Based on **Definition 2**, let us consider the two groups g_1 ={p_1(3,3), p_2(0,4)} and g_7 ={p_2(0,4), p_4(2,3)}. Since here the greatest value, the better, one can check that g_1 g-dominates g_7. Indeed, one can find a permutation for g_1 given by {p_2(0,4), p_1(3,3)} such that p_2(0,4) $\succeq$ p_2 (0,4) and p_1(3,3) $\succ$ p_4(2,3) (where $\succeq$ and $\succ$ are respectively *the preferred-or-equal relation* and *Pareto dominance relation* based on the order relations $\geq$ and $>$). Therefore, g_7 is not a skyline group. The same process leads us to a set of skyline group {g_1, g_2, g_3, g_6}, these are the only skyline groups as no other group with 2 points can g-dominate g_1, g_2, g_3 and g_6.

Based on the experimental evaluation done in [41], it has been proved that it is more interesting to consider **Definition 1** because monotone function definition is a subset of the permutation definition and generally it returns less information (i.e., a small number of skyline groups) compared to **Definition 2**. In the rest of the paper, we make use of the **Definition 1** when computing the group skyline. ■

Fuzzy Set Theory: A refresher The first article in fuzzy set theory written by Zadeh in 1965 [36] shows the intention of the author to generalize the classical notion of a set and a proposition to accommodate fuzziness to represent classes or sets of objects with all-defined boundaries. These sets allow us to describe gradual transitions between total membership and absolute rejection. Typical examples of these fuzzy classes are those described using adjectives or adverbs of the natural language, such as not cheap, young and tall. Formally, a fuzzy set F on the universe X is described by a membership function μ_F: X $\rightarrow$ [0, 1], where μ_F(x) represents the degree of membership of x in F.

Using this definition, if μ_F(x)=0 then the element x $\notin$ F, if μ_F(x) = 1 then x $\in$ F, these elements represent the core of F denoted by CORE(F) = {x $\in$ F | μ_F(x) = 1}. When $0 < \mu_F(x) < 1$, it became a partial membership, these elements form the support of F denoted by SUPP(F) = {x $\in$ F | μ_F(x)$>$ 0}. The complement of F, denoted $\overline{F}$, is defined by $\mu_{\overline{F}}(x)$

= 1 - $\mu_F(x)$. More μ_F(x) is close to the value 1, more x belongs to F. Therefore, given x, y ∈ F, we say that x is preferred to y iff μ_F(x) > μ_F(y). If μ_F(x) = μ_F(y), then x and y have the same preference.

In practice, F can be represented by a trapezoid membership function (t.m.f) $(\alpha, \beta, \varphi, \psi)$, where $[\beta, \varphi]$ is the core and $]\alpha, \psi[$ is its support (see Figure 2). This kind of membership functions in addition to its simple representation (only a quadruplet of values is need), leads to uncomplicated computational operations as well.

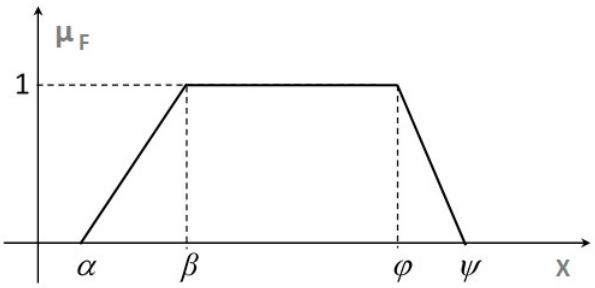


Fig. 2. Trapezoidal membership function

2.2. Problem description

Let $\mathbb{Q}$ be a user skyline query and $\mathbb{D}$ be the target dataset. Assume the user wants to find K groups of l elements for decision purposes. Let G-SKY($\mathbb{D}$) be the group skyline computed and | G-SKY($\mathbb{D}$) | be its size.

One can easily observe that if | G-SKY($\mathbb{D}$) | < K, the user is not then able to make the desired decision due the insufficient skyline groups returned. The problem of interest is then how to enlarge the size of G-SKY($\mathbb{D}$) in order to obtain a relaxed variant, called RG-SKY($\mathbb{D}$), with more groups (i.e., G-SKY($\mathbb{D}$) ⊆ RG-SKY($\mathbb{D}$)). We call this problem the group skyline relaxation problem.

3. Related work

We review here the main research works related to group skyline both from the computation and semantics point of view. We also provide a comprehensive comparison of those works w.r.t. to a set of criteria conveniently chosen.

The brute method to get G-SKY is to enumerate all the groups, and then run the query based on the group dominance relationship. The brute force method computes the aggregate tuple for each group, then uses any traditional skyline algorithm to find the group skyline. This method is significantly time consuming and the storage cost can be exponential due to the huge intermediate input for the traditional skyline tuple algorithm. In [38] some alternatives to this naive method are proposed. The existing group skyline propositions focus on the (i) problem using one of the definitions presented in Section 2 on stream or static data, e.g,. the papers [15,29,32,33,37,40,43] combine the advantages of group skyline and top-k queries. Another work [19] presents a structure that represents the points in a directed skyline graph and captures all the dominance relationship among the

points based on the notion of skyline layers. Some papers [8,12,17] focus on stream data to compute G-SKY continuously, where they invoke the problem of computing G-SKY when a new point p arrives dynamically. The underlying idea is to store dominance information that could be reused in another search space pruning. In [14,18], authors generate candidate groups in a progressive manner and update the resulting group skyline dynamically. Some other papers [6,7,13,34,35,39,41,42,44] focus on optimizing experimental performance of the group skyline algorithm by using Parallelization and other methods. Based on our previous survey papers [21,23], we summarize and compare the above existing works in Table 4 (where (i) **D.type (Data type):** stands for stream or static data, (ii) **Perf (Performance):** means that the work focuses on the execution time rather than the quality of the responses, (iii) **Def (Definition):** indicates which definition 1 or 2 used to compute the group skyline, (iv) **Ref (Refinement):** means that the work is endowed with a refining process to reduce the skyline groups set returned and (v) **Rel (Relaxation):** means that the work tries to get more skyline groups by relaxing the definition of the traditional skyline groups).

As it can be seen, none of the previous work deals with the silence problem (Namely, the set of answers is empty or insufficient to decision making) in the group skyline context compared to the traditional skyline where we can cite the two papers [2,20] that use respectively the kNN definition and a fuzzy preference relation to relax the skyline result.

Table 4. Group Skyline Related Work comparison

Related Work	D.type	Perf.	Def.	Ref.	Rel.
Efficient computation of combinatorial skyline queries [6]	Static	✓	1	✓	✗
An Efficient Algorithm to Compute Compositional Skyline[7]	Static	✓	None	✓	✗
Finding Group-Based Skyline over a Data Stream in the Sensor Network[8]	Stream	✓	2	✓	✗
Efficient processing of skyline group queries over a data stream[12]	Stream	✓	2	✓	✗
Combination skyline queries[13]	Static	✓	1	✓	✗
Group skyline computation[14]	Static	✓	2	✓	✗
Incremental evaluation of top-k combinatorial metric skyline query[15]	Static	✓	None	✓	✗
Progressive approaches for Pareto optimal groups computation[17]	Stream	✓	None	✓	✗
Discovering Group Skylines with Constraints by Early Candidate Pruning[18]	Static	✓	1	✓	✗
Finding pareto optimal groups: Group-based skyline[19]	Static	✓	1	✓	✗
Top-k combinatorial skyline queries[29]	Static	✓	1	✓	✗
Identifying Most Preferential Skyline Product Combinations[32]	Static	✓	None	✓	✗
Identifying most preferential skyline product combinations under price promotion[33]	Static	✓	None	✓	✗
Efficient Contour Computation of Group-based Skyline[34]	Static	✓	2	✓	✗
Fast algorithms for pareto optimal group-based skyline[35]	Static	✓	2	✓	✗
Finding k-Dominant G-Skyline Groups on High Dimensional Data[37]	Static	✓	None	✓	✗
On skyline groups[38]	Static	✓	2	✓	✗
Finding optimal skyline product combinations under price promotion [39]	Static	✓	None	✓	✗
Top-k Dominating Queries on Skyline Groups[40]	Static	✓	1,2	✓	✗
Computing skyline groups: an experimental evaluation[41]	Static	✓	1,2	✓	✗
Computing Skyline Groups: An Experimental Evaluation [42]	Static	✓	1,2	✓	✗
Top-k Skyline Groups Queries[43]	Static	✓	1	✓	✗
Parallelization of group-based skyline computation for multi-core processors[44]	Static	✓	2	✓	✗

✓ Yes ✗ No

4. Group skyline relaxation approach

We discuss here our fuzzy approach to relax the group skyline. The idea is to extend the group dominance (given in Definition 1) by making it more demanding. The relaxed group Skyline obtained, RG-SKY, is no longer a flat set but a discriminated set where each of its elements is associated with a degree.

The main idea consists of computing the extent to which a group, discarded by the G-Skyline dominance relationship (denoted $\prec_{gs}$, see Definition 1), may belong to the relaxed group skyline. To this end, and as it will be illustrated further, we associate with each skyline attribute A_i ($i \in \{1, \cdots, d\}$) a pair of parameters $(\gamma_{i1}, \gamma_{i2})$ where γ_{i1} and γ_{i2} respectively denote the bounds of the relaxation zone allowed to the attribute A_i. A vector of pairs of parameters, denoted γ, is then defined as

$$\gamma = ((\gamma_{11}, \gamma_{12}), \cdots, (\gamma_{d1}, \gamma_{d2})).$$

It is worthy to note that γ, called a relaxation parameter vector, is a user-defined[6]. It defines the set of values w.r.t each attribute that user can tolerate despite they are ruled out when applying the dominance $\prec_{gs}$.

4.1. Fuzzy group dominance

RG-SKY, the relaxed group skyline of G-SKY, relies on a particular dominance relationship (inspired from the work [2]) that allows enlarging the group skyline with the most interesting groups among those ruled out when computing G-SKY using Definition 1. This dominance relationship makes use of the fuzzy relation "Much Preferred ($MP_{\mathbb{G}}$)" to compare two groups g and g'. So, g is an element of RG-SKY if there is no group g' $\in \mathbb{G}$ such that g' is much preferred to g (denoted $MP_{\mathbb{G}}$(g',g)) in all group skyline attributes. Formally, we write:

$$g \in RG - SKY \Leftrightarrow \nexists g' \in \mathbb{G}, MP_{\mathbb{G}}(g', g) \tag{4}$$

Note that g' is much preferred to g in the sense of $MP_{\mathbb{G}}$ if and only if g' is much preferred to g w.r.t. to all group skyline dimension i in $\{1, ..., d\}$. Formally, we write:

$$MP_{\mathbb{G}}(g', g) \Leftrightarrow \forall i \in \{1, ..., d\}, MP_{\mathbb{G}_i}(g'(i), g(i)) \tag{5}$$

where $MP_{\mathbb{G}_i}$ is a defined on the domain D_i of the attribute A_i, $g(i) = F(Q_1[i], ..., Q_l[i])$ (resp. $g'(i) = F(Q'_1[i], ..., Q'_l[i])$ and F is an aggregate function. Recall that $(g'(i), g(i)) \in MP_{\mathbb{G}_i}$ means that $\mu_{MP_{\mathbb{G}_i}}(g'(i), g(i)) > 0$ (or $MP_{\mathbb{G}_i}(g'(i), g(i)) > 0$ for short). In a similar way, $(g', g) \in MP_{\mathbb{G}}$ means also $\mu_{MP_{\mathbb{G}}}(g', g) > 0$ (or $MP_{\mathbb{G}}(g', g) > 0$ for short).

Note that $MP_{\mathbb{G}_i}(g'(i), g(i))$ expresses the extent to which the value $g'(i)$ is much preferred to the value $g(i)$. Since $MP_{\mathbb{G}_i}$ is of a gradual nature, each element g of RG-SKY is associated with a degree ($\in$[0,1]) expressing the extent to which g belongs to RG-SKY. Now in fuzzy set terms, one can write equation (4) as follows (where the quantifiers $\forall$ and $\exists$ are modeled by the min and max operators respectively):

[6] The user predefine the values or the degree of tolerance of each dimension in a form of a vector of relaxation.

$$\mu_{RG-SKY}(g) = 1 - \max_{g' \in \mathbb{G}-\{g\}} \min_i \mu_{MP_{\mathbb{G}_i}}(g'(i), g(i)) = \min_{g' \in \mathbb{G}-\{g\}} \max_i (1 - \mu_{MP_{\mathbb{G}_i}}(g'(i), g(i))) \tag{6}$$

The semantics of the fuzzy relation $MP_{\mathbb{G}_i}$ can be expressed by the following formulas (7) (see also figure 3).

$$\mu_{MP_{\mathbb{G}_i}^{(\gamma_{i1},\gamma_{i2})}}(g'(i), g(i)) = \begin{cases} 0 & \text{if } g'(i) - g(i) \leq \gamma_{i1} \\ 1 & \text{if } g'(i) - g(i) \geq \gamma_{i2} \\ \frac{(g'(i)-g(i))-\gamma_{i1}}{\gamma_{i2}-\gamma_{i1}} & \text{elsewhere} \end{cases} \tag{7}$$

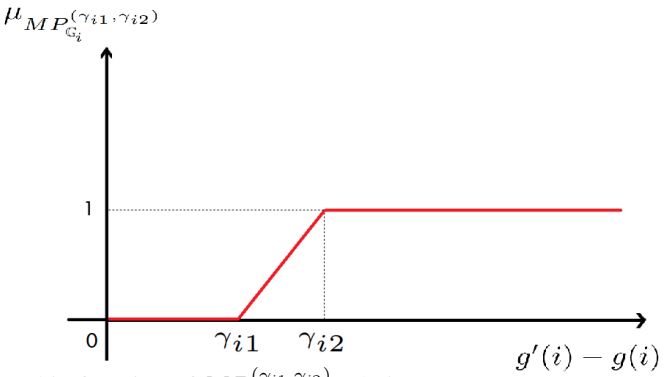


Fig. 3. Membership function of $MP_{\mathbb{G}_i}^{(\gamma_{i1},\gamma_{i2})}$ relation

For instance, if $g'(i) - g(i) \geq \gamma_{i2}$ then $g'(i)$ is completely *much preferred* to $g(i)$. One can also see that if $g'(i) - g(i) > \gamma_{i1}$, $g'(i)$ is not only *preferred* but *much preferred* to $g(i)$ to some extent. In terms of t.m.f., the fuzzy set associated with $MP_{\mathbb{G}_i}$ writes $(\gamma_{i1}, \gamma_{i2}, \infty, \infty)$, and denoted $MP_{\mathbb{G}_i}^{(\gamma_{i1},\gamma_{i2})}$. It is easy to check that $MP_{\mathbb{G}_i}^{(0,0)}$ corresponds to the crisp preference relation expressed by means of the regular relation *"greater than"*.

Now, let $RG\_SKY^{(\gamma)}$ be the relaxed group skyline computed on the basis of the relaxation vector $\gamma = ((\gamma_{11}, \gamma_{12}), \cdots, (\gamma_{d1}, \gamma_{d2}))$ in the case of d skyline attributes. One can easily check that the group skyline $G - SKY = RG\_SKY^{(\mathbf{0})}$ with $\mathbf{0} = ((0,0), \cdots, (0,0))$.

One can also check that the following monotonicity property holds.

Proposition 1. *Let γ and γ' be two relaxation parameter vectors. Then, the following propriety holds:*

$$\gamma' \leq \gamma \Rightarrow RG - SKY^{(\gamma')} \subseteq RG - SKY^{(\gamma)}$$

Proof. (Sketch) Let $\gamma = ((\gamma_{11}, \gamma_{12}), \cdots, (\gamma_{d1}, \gamma_{d2}))$ and $\gamma' = ((\gamma'_{11}, \gamma'_{12}), \cdots, (\gamma'_{d1}, \gamma'_{d2}))$ two relaxation parameter vectors. $\gamma' \leq \gamma \Rightarrow \forall i \in \{1, ..., d\}\ \gamma'_{i1} \leq \gamma_{i1}$ and $\gamma'_{i2} \leq \gamma_{i2}$. This implies that $\forall i \in \{1, ..., d\}\ MP_{\mathbb{G}_i}^{(\gamma_{i1},\gamma_{i2})} \subseteq MP_{\mathbb{G}_i}^{(\gamma'_{i1},\gamma'_{i2})}$. Based on (6), one can deduce that $RG - SKY^{(\gamma')} \subseteq RG - SKY^{(\gamma)}$ holds.

Lemma 1. *Let $\gamma = ((0, \gamma_{12}), \cdots, (0, \gamma_{d2}))$, $\gamma' = ((\gamma'_{11}, \gamma'_{12}), \cdots, (\gamma'_{d1}, \gamma'_{d2}))$ and $\forall\ i \in \{1, ..., d\}\ \gamma_{i2} < \gamma'_{i2}$, the following result holds as well:*

$$RG - SKY^{(0)} \subseteq RG - SKY^{(\gamma)} \subseteq RG - SKY^{(\gamma')}$$

Table 5. Hotels conference example

Hotels	Price (Euro)	Distance (Km)
h_1	50	10
h_2	100	6
h_3	150	5
h_4	40	11

Example 3. (Continued) To illustrate the interest of the RG-SKY set, let us consider a simple example of 4 hotels with close values as depicted in Table 5. Assume the user wants to retrieve the K = 3 best groups of hotels by minimizing the two dimensions price and the distance, to get the closest hotels and the cheapest.

To this end, we proceed as follows:

1. Generate the skyline points: It is easy to check SKY = $\{ h_1, h_2, h_3, h_4\}$ because they are incomparable.
2. Generate the groups of size 3 using the binomial coefficient[7] and apply the MIN aggregation function (F_{min}):
   $g_1 = \{h_2, h_3, h_4\}$, $F_{min}(g_1)$=<min(100,150,40), min(6,5,11)>=<40, 5>
   $g_2 = \{h_1, h_3, h_4\}$, $F_{min}(g_2)$=<min(50,150,40), min(10,5,11)>=<40, 5>
   $g_3 = \{h_1, h_2, h_4\}$, $F_{min}(g_3)$=<min(50,100,40), min(10,6,11)>=<40, 6>
   $g_4 = \{h_1, h_2, h_3\}$, $F_{min}(g_4)$=<min(50,100,150), min(10,6,5)>=<50, 5>
3. By applying Definition 1, one can check that the group skyline is
   G-SKY = $\{g_1, g_2\}$.
   Unfortunately, the user receives only 2 skyline groups even if all tuples are skyline points while (s)he needs 3 groups to make a decision. To satisfy the user's needs, we call then the RG-SKY method.

RG-SKY method:
Let us first assume that the relaxation vector $\gamma = ((0.5, 1), (0.5, 1))$, i.e. $(0.5, 1)$ both for the skyline attributes "Price" and "Distance". According to equation (7), one can check that the fuzzy relation $MP_{\mathbb{G}_{Price}}$ can write:

$$\mu_{MP^{(0.5,1)}_{\mathbb{G}_{Price}}}(v, u) = \begin{cases} 0 & \text{If } v - u \leq 0.5 \\ 1 & \text{If } v - u \geq 1 \\ \frac{v-u-0.5}{1-0.5} & \text{Otherwise} \end{cases} \quad (8)$$

The fuzzy relation $MP_{\mathbb{G}_{Distance}}$ can also be written in a similar way.
Let us now compute the fuzzy set RG-SKY using equation (6) (where $i = 1$ and $i = 2$ denote the attributes *Price* and *Distance* respectively):

$\mu_{RG-SKY}(g_3) = 1 - \max\limits_{g' \in \{g_1, g_2, g_4\}} \min\limits_{i \in \{1,2\}} \mu_{MP_{\mathbb{G}_i}}(g'(i), g_3(i))$
$\mu_{RG-SKY}(g_3)$ = 1 - max[min($\mu_{MP_{\mathbb{G}_1}}(g_1(1), g_3(1))$,$\mu_{MP_{\mathbb{G}_2}}(g_1(2), g_3(2))$), $min(\mu_{MP_{\mathbb{G}_1}}(g_2(1),$ $g_3(1))$, $\mu_{MP_{\mathbb{G}_2}}(g_2(1), g_3(2)))$, $min(\mu_{MP_{\mathbb{G}_1}}(g_4(1), g_3(1))$, $\mu_{MP_{\mathbb{G}_2}}(g_4(2), g_3(2)))$]

[7] The binomial coefficient is noted $C(n, k)$ and reads choose k among n and is defined by the formula $C(n, k) = n!/(k!(n - k)!)$ with $n!$ stands for the factorial of n.

$\mu_{RG-SKY}(g_3)$ = 1 - max[min(0, 0), min(0, 0), min(1, 0)] = 1 - 0 = 1

In a similar way, we obtain $\mu_{RG-SKY}(g_4) = 1$. Then,

RG-SKY = $\{1/g_1, 1/g_2, 1/g_3, 1/g_4\}$.

One can observe that RG-SKY contains the G-SKY elements (i.e., g_1 and g_2 with a degree equals 1) and some new groups that were not in G-SKY with a degree equals 1 (i.e., g_3 and g_4). RG-SKY is more larger than G-SKY and can satisfy the initial user query by returning for instance the groups g_1, g_2 and g_3.
Now, if RG-SKY contains more than K groups, the K best groups are returned. In case of ties, the user can establish a rank-order on the basis of the preferences w.r.t. the skyline attributes (in our case, the *Price* and *Distance* attributes). As for the case where RG-SKY contains less than K groups, one can revise the relaxation vector and re-execute the RG-SKY method. ■

4.2. Some basic properties

We establish here a set of desirable properties that are of interest for computation purpose. Some of them are the fuzzy counterparts of group skyline proprieties introduced in [14]. Let $\gamma = ((\gamma_{11}, \gamma_{12}), \cdots, (\gamma_{d1}, \gamma_{d2}))$ and $\gamma' = ((\gamma'_{11}, \gamma'_{12}), \cdots, (\gamma'_{d1}, \gamma'_{d2}))$ be two relaxation parameter vectors (where $MP^{\gamma}_{\mathbb{G}}(g, g')$
> 0 means $(g, g') \in MP^{\gamma}_{\mathbb{G}}$):

Proposition 2. *(Min-Asymmetry) Let g and g' be two groups of* $\mathbb{G}$*,*

$$\textit{If } (g, g') \in MP^{\gamma}_{\mathbb{G}} \textit{ then } (g', g) \notin MP^{\gamma}_{\mathbb{G}}.$$

Proof. Proposition 2 can also be written in the form: If $MP^{\gamma}_{\mathbb{G}}(g, g') > 0$ then $MP^{\gamma}_{\mathbb{G}}(g', g)$ $= 0$. Now, due to the asymmetry property of the fuzzy preferences [28], one can write: $min(MP^{(\gamma_{i1}, \gamma_{i2})}_{\mathbb{G}_i}(g(i), g'(i)), MP^{(\gamma_{i1}, \gamma_{i2})}_{\mathbb{G}_i}(g'(i), g(i))) = 0, \forall i \in \{1, ..., d\}$. Namely, if $MP^{(\gamma_{i1}, \gamma_{i2})}_{\mathbb{G}_i}(g(i), g'(i)) > 0$ then $MP^{(\gamma_{i1}, \gamma_{i2})}_{\mathbb{G}_i}(g'(i), g(i)) = 0$.

Proposition 3. *(Min-Transitivity) Let g, g' and g" be three groups of* $\mathbb{G}$*,*

$$\textit{If } (g, g') \in MP^{\gamma}_{\mathbb{G}} \textit{ and } (g', g") \in MP^{\gamma'}_{\mathbb{G}} \textit{ then } (g, g") \in MP^{\gamma+\gamma'}_{\mathbb{G}}.$$

Proof. Proposition 3 can writes also in the form: If $MP^{\gamma}_{\mathbb{G}}(g, g') > 0$ and $MP^{\gamma'}_{\mathbb{G}}(g', g") >$ 0 then $MP^{\gamma+\gamma'}_{\mathbb{G}}(g, g") > 0$. Now, due to the transitivity property of the fuzzy preferences [28] and to the fuzzy addition formula [9], one can write:
$min(MP^{(\gamma_{i1}, \gamma_{i2})}_{\mathbb{G}_i}(g(i), g'(i)), MP^{(\gamma'_{i1}, \gamma'_{i2})}_{\mathbb{G}_i}(g'(i), g"(i))) \leq MP^{(\gamma_{i1}+\gamma'_{i1}, \gamma_{i2}+\gamma'_{i2})}_{\mathbb{G}_i}(g(i),$
$g"(i)), \forall i \in \{1, ..., d\}$.

Proposition 4. *If g* $\subset$ *SKY(*$\mathbb{D}$*) then g* $\in$ *RG-SKY(*$\mathbb{D}$*) does not always hold.*

Proof. To show that this Proposition is not always true, it suffices to exhibit a counterexample. Consider a 2-dimensional (points, rebounds) dataset of 4 players: p_1 = (0,4), p_2 = (1,2), p_3 = (2,1), and p_4 = (4, 0). It is easy to see that $SKY = \{p_1, p_2, p_3, p_4\}$. Let $g = \{p_1, p_4\}$ ($\subset$ SKY). One can check that $g \notin$ RG-SKY for $\gamma = ((0, 1), (0, 1))$.

Proposition 5. *(The converse of Proposition 4) If $g \in$ RG-SKY($\mathbb{D}$) then $g \subset$ SKY($\mathbb{D}$) does not always hold.*

Proof. Let us also find a counterexample. Consider a 2-dimensional (points, rebounds) dataset of 3 players: p_1 = (0, 2), p_2 = (1, 0), and p_3 = (2,1), we have SKY=$\{p_3\}$ and G-SKY=$\{\{p_1, p_3\}\}$ (i.e. contains 1 group of 2 points). For $\gamma = ((0,1),(0,1))$, one can check that $\{p_2, p_3\} \in RG - SKY$ while $\{p_2, p_3\} \notin SKY$.

Table 6. Comparison of the general Skyline algorithms

Name	Indexed	Limits and issues
Index [30]	✓	- Does not support user-defined preferences (the order of the returned points is fixed and depends on the distribution of the data values)
Bitmap [30]	✓	- The memory consumption limit due to the conversion of points to Bitmap structure - Bitmap also handles inefficiently updates because it implies the recalculation of all the bit vectors t - Does not work on several dimensions (expensive) - Does not allow the user to express preferences - Mandatory to code all the point values
NN [16,31]	✓	- Performance problem, in case, we have a single element, the algorithm continue the division to 4 regions whereas the program can check in advance the number of the existing points - It is not efficient if the data is not mass
BBS [25,27,31]	✓	- Does not work when dimensions exceed 5
BNL [3,31]	✗	- Requires a lot of iterations before the final skyline is calculated (it analyzes all the data) - It has a limit on the size of the window, the complexity of the algorithm depends on this size - A non-negligible comparison time is necessary - The skyline points are not defined progressively (they change)
D&C [3]	✗	- The skyline points are not defined progressively - Problem if the data is so small (the process becomes useless), it is more efficient when dealing with a large amount of data - The algorithm scans the entire database
SFS [5]	✗	- Scans all data - The necessity to define a good monotonous function
LESS [10]	✗	- Elimination-filter (EF) mechanism can become full - All the data must be scanned at least once

5. RG-SKY computation

RG-SKY computation is expected to be a part of a Decision Support System (DSS) [22]. Here, we provide (i) a diagram which gives an overview of how the RG-SKY approach works and (ii) the two proposed algorithms for computing the RG-SKY set.

5.1. RG-SKY diagram

Figure 4 provides an overview of the RG-SKY approach and illustrates the chronology of its steps in a comprehensive way. Two cases can be distinguished:

– **Case 1:** User request satisfaction - Relaxation unneeded
In this case, the decision maker sends a request with a set of conflicting conditions to the DSS system (that integrates the RG-SKY computation process), the traditional G-SKY computation returns a satisfactory answer and the RG-SKY process is then not triggered.

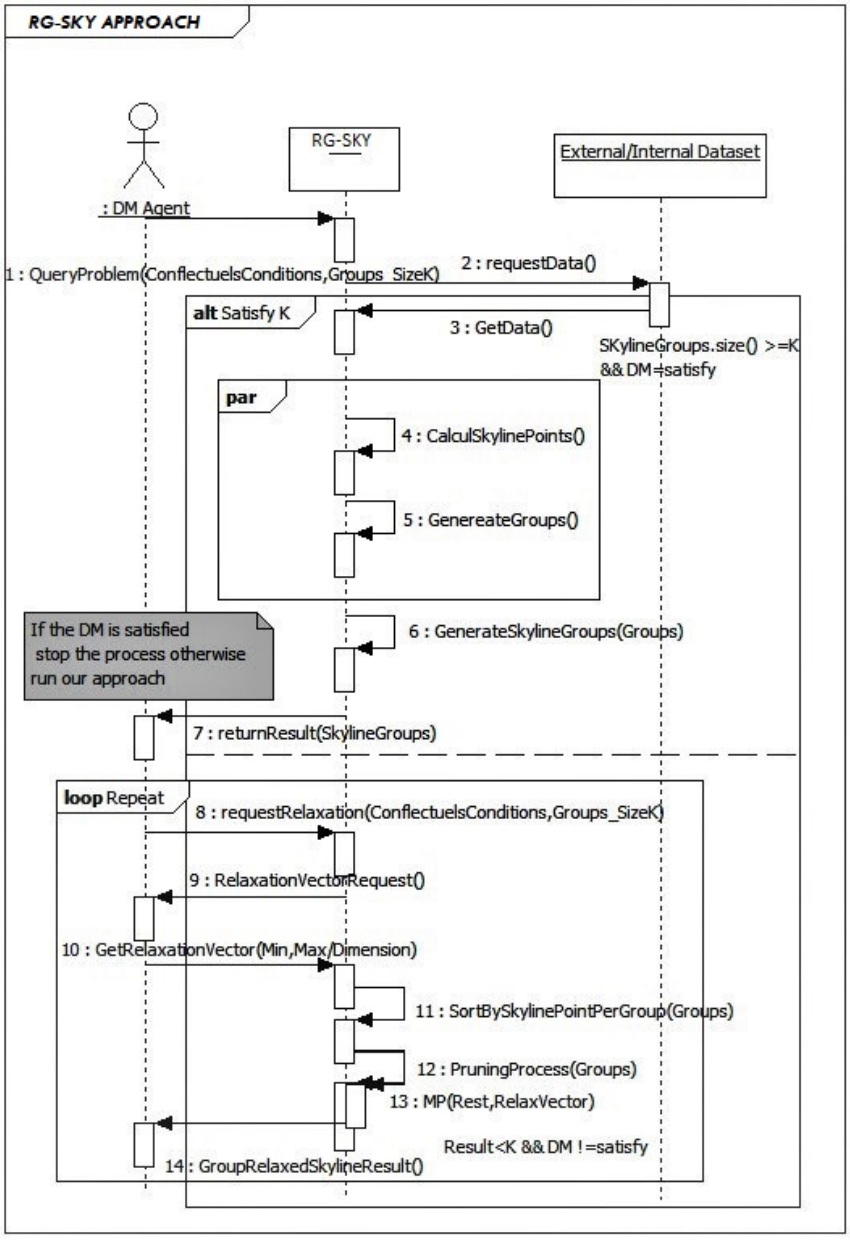


Fig. 4. Sequence diagram of the general RG-SKY approach

– **Case 2:** User request dissatisfaction - Relaxation needed
The traditional G-SKY computation returns an unsatisfactory answer. Before starting the relaxation process, the system first gets the relaxation parameter vector from the user. The system also executes some pre-processing and some pruning techniques to speed-up the RG-SKY computation. For instance, the following optimization strategies are implemented:

  - **Sorting groups by their number of skyline points:** this allows creating an hierarchy of the groups that helps intelligently to scan the research space.
  - **Pruning techniques:** they are based on the previous established propositions and properties.

  The RG-SKY computed represents a fuzzy set in the sense that each group is associated with a degree (expressing to what extent is not "much dominated" by any other group). If RG-SKY contains more K groups, the top-K groups are returned to the user and then the process stops. Otherwise, we revise the relaxation parameter vector (making it more permissive) and we re-compute the RG-SKY.

RG-SKY naive algorithm This subsection presents the first algorithm (Algorithm 1) proposed to implement the RG-SKY approach. It does not use any optimization technique to reduce the research space.

Algorithm 1: RG-SKY naive algorithm

Result: Relaxed Group Skyline
Input : G-SKY: Skyline groups, $\mathbb{G}$: Groups, K: integer, γ: relaxation Vector
Output: RG-SKY

```
RG-SKY ← G-SKY                    // The first step is the generation of G-SKY
GetRestGroups(G-SKY,𝔾);
SortbySkyPoints(G-REST);          // pre-treatment phase
DeleteZeroSkyGroups(G-REST);      // pruning phase
j ⟵ 1;
while (G-REST.length ≤ k) do
    // level: denotes the number of skyline points of the current group
    for(i=0;i<N;i++){
    Compile μ_MP (level[j].get(i),γ);      // g'∈Rest, μ_MPi(g'i,gi)
    j++;
end
+ Return RG-SKY
```

RG-SKY Salsa algorithm This algorithm is an improved version of the naive algorithm where we avoid analyzing all the groups generated to extract the skyline groups. This choice relies on the comparative study conducted on a set of well-known skyline algorithms, as explained below.

Algorithm 2: RG-SKY SALSA algorithm

Input : G-SKY: Group Skyline, $\mathbb{G}$: Groups, k: integer, γ: relaxation Vector, F: A monotone sorting function
Output: RG-SKY: Relaxed Group Skyline

```
1  RG-SKY ← G-SKY
2  GetRestGroups(G-SKY,𝔾);          // G-Rest
3  SortbySkyPoints(G-Rest);
4  DeleteZeroSkyGroups(G-REST);
5  RG-SKY ⟵ G-SKY; stop ⟵ false; pstop ⟵ undefined;
6  Sort G-REST according to F;      // F(G-REST);
7  while (¬ pstop and G-REST ≠ ∅ ) do
8  |  G ⟵ get next group from G-REST;
9  |  G-REST⟵ G-REST\{G}
10 |  if ¬ MP(G,G-SKY) then RG-SKY ⟵ RG-SKY ∪ {G}, update pstop then
11 |  if pstop > G-SKY then
12 |     stop := true;
13 end
14 Return RG-SKY;
```

To integrate the RG-SKY approach in our Decision Support System Model proposed in [22], knowing that group skyline query processing results in an expensive procedure, it is then important to choose the best adaptive skyline algorithm for our context. The choice of SaLSa (Sorting and Limit skyline algorithm) is justified by the fact that it overcomes the main limitations found in the other general skyline algorithms, as summarized in Table 6. In this comparative study, we have considered only the well-known algorithms in the skyline field. For a complete overview on skyline algorithms, see for instance [24,26].

SaLSa algorithm used in the traditional skyline query extraction, is an improvement of SFS and LESS (see Table 6). It strives to avoid scanning the entire sorted dataset as opposed to the previous propositions, it is the first algorithm that exploits the values of a monotonic notation (limitation) function to sort the data set to read and compare. SaLSa differs from the other generic algorithms because it consistently limits the number of points read and the dominance tests. The design of SaLSa is based on two key concepts: First, a sorting step of the input data and, second, suitably choose a sorting function that does not privilege any attribute over the others (the function does not influence the correctness of SaLSa but only its performance). For these reasons, we adapt the Salsa algorithm to the group skyline problem and to optimize also the relaxation process of the RG-SKY approach. During the filter phase, the algorithm reads and examines the rest of the groups. Each time a new group is read, it is compared to the current skyline group list. If a group dominates the current group, it is ignored, otherwise it is inserted to the relaxed skyline groups list (as a final relaxed group skyline) and the algorithm checks its termination trigger (Pstop). If the current threshold Pstop is less than or equal to the *fmin* value of the point, the algorithm ends and returns the entire skyline RG-SKY group list. This termination condition ensures that no data groups examined later should be part of the RG-SKY list, thereby the algorithm avoids analyzing the entire rest of the dataset.

There are many limiting functions in the literature, but the optimal function that can limit any input relation more than others do, is the Minimum Coordinate Function (MCF) comparing to Sum and Val presented in [1] (MCF is noted MinC, first it sorts groups considering the minimum coordinate value of the current group and simultaneously Sum function of group elements is calculated and used in case of ties).

Table 7. Set of parameters (where N stands for the number of input groups)

Parameter	Values	default value
Groups [N]	NBA={500}, HOUSE={5911}, WEATHER={4438}, Correlated={10,50,100,500,2000-3612281}, Anti-correlated={2700}, Independent={1500}	10 50 100 500 1000
Dataset distribution schema	Correlated, Independent, Anticorrelated	Correlated
Number of group dimensions [l]	2, 3, 4, 5	2
Relaxation vector [(γ_1, γ_2)]	$\gamma_1 \in [0, 1], \gamma_2 \in [0, 1]$	(0, 1)

6. Experimental study

Table 8. Specifications of real datasets

Dataset	Cardinality	Dimensionality
NBA	17,264	8
House	127,931	6
Weather	566,268	15

This section presents the experimental study carried out. It validates the effectiveness and the relevance of the RG-SKY approach to relax small group skylines and also measures some performance related to the computational time.

6.1. Experimental environment

The algorithms are tested on a Dell Inc Machine, System Model: Precision T1650 and run in a Windows 10 Education 64-bit (10.0, Build 16299) environment, using a 3.4GHz Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz(8 CPUs) with a main memory of 8GB RAM, in sequential mode (1 thread) and a 500 GB of disk. Dataset benchmark is generated using the method described in [3] following three conventional distribution schema (correlated, anti-correlated and independent) and also ”randataset”. The approach was developed in Eclipse Modeling Tools Version: Oxygen.3 Release (4.7.3), using Java.v9 language.

6.2. Experimental tests

The tests can be classified into two main parts: Real Data (NBA, HOUSE, WEATHER) and Synthetic data where we change the data type and size, the groups and tuples dimensions and finally generate data for the purpose of the relaxation parameter vector tests

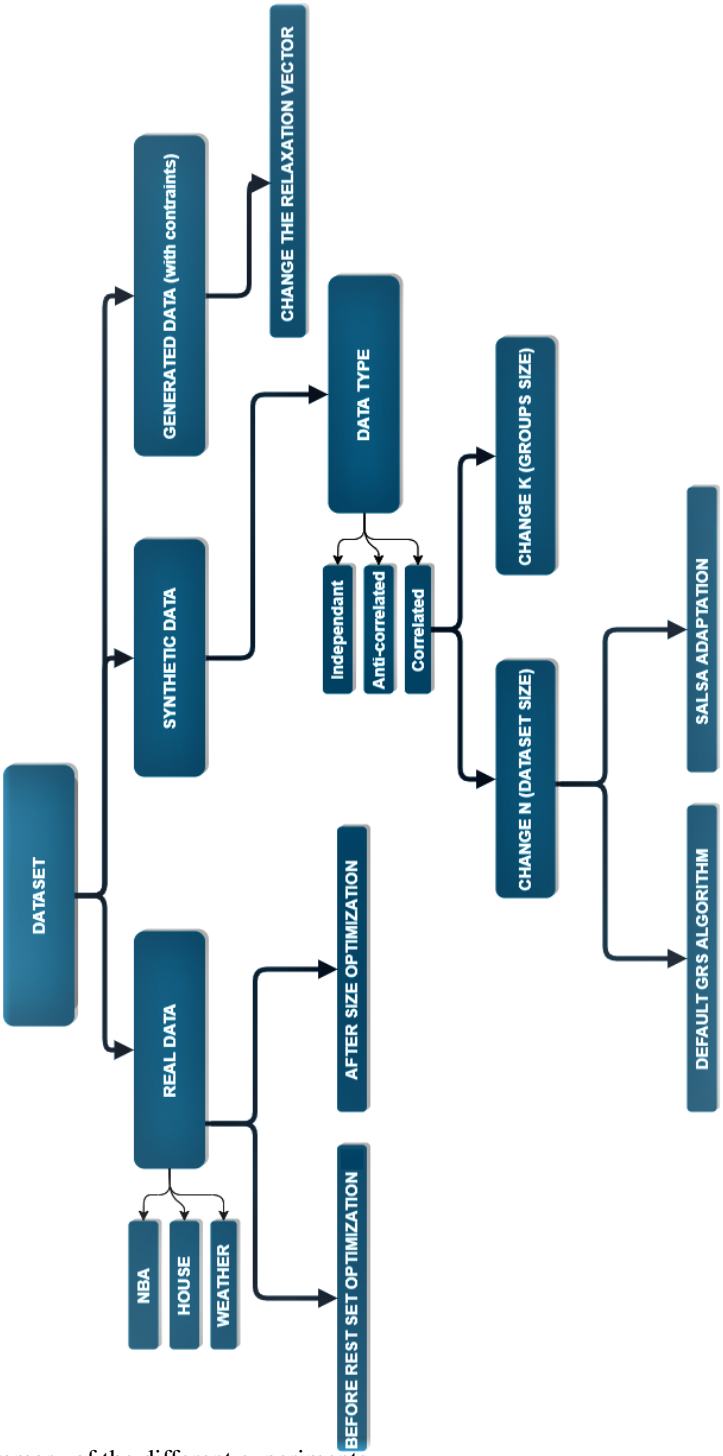


Fig. 5. Summary of the different experiments

(The choice a very low values can motivate our approach and prove the utility of getting results even if the user-defined values are small). Existing skyline researchers [14,38] use the data generator proposed in [3] with standard parameters so that results can be directly compared. As our work targets the same kind of data, we propose to follow the same methodology to evaluate our proposal. The data generator in Borzsonyi et al. (2001) generates database tuples (or points) with varying numbers of dimensions (or attributes). Tuples are generated using one of the following three value distributions: – correlated (corr): tuples, which are good in one dimension, tend to be good in other dimensions, too; – anti-correlated (anti): tuples, which are good in one dimension, are bad in at least one other dimension; –independent (indep): tuples are generated using a uniform distribution. We also include three real datasets (see Table 8 for the specification of those datasets) that are commonly used to evaluate skyline algorithms [4]: NBA (statistics of basketball players during regular seasons), HOUSE (money spent in one year by an American family for six different types of expenditures) and WEATHER (average monthly precipitation totals and elevation at over half a million sensor locations). Finally, a last test is done on generated real data using Skyline generator (randdataset). We summarize our input values (constant/variable) in table 7 and our experimental tests in figure 5. Note that in all our experiments, we make use of the aggregation function "MIN". Since this function returns less skyline groups than other functions such MAX and SUM (as shown in reference [38]). This behavior of the "MIN" function is more interesting for the relaxation purpose.

6.3. Experimental results

Case 1: Real Data
Figure 6 shows the number of relaxed skyline groups in a different data type and the execution time of the RG-SKY approach using the adapted aggregation function RG-SKY provides more groups comparing to the set G-SKY. One can observe that for the three datasets (NBA, WEATHER, HOUSE) G-SKY contains only one group.

The execution time depicted in Figure 6 and resulting from Algorithm 1 is not similar for the three datasets due to their different correlations and sizes, while this time is similar for the (correlated) NBA dataset.

After this first execution, we propose an optimized algorithm (Algorithm 2) that leads to 97.20% improvement of the naive version (i.e., Algorithm 1) in terms of execution time. This why for all the next experiments, Algoritm 2 is used.

Case 2: Synthetic Data (Correlated)

– **Date type variation** Figure 7 shows the execution time and the returned number of relaxed skyline groups in different data types: correlated, anti-correlated and independent data. As can be seen, the set G-SKY always contains one group for the three datasets compared to RG-Sky which returns more than one group, except for correlated data that returns the same number of skyline groups.
On the other hand, one can observe that independent and anti-correlated data are time consuming compared to the correlated data. We note also that our RG-SKY Salsa algorithm (Algorithm 2) is equal or less time consuming compared to the existing G-SKY naive algorithm (Except for anti-correlated data).
This is why we decide to continue our experiments only on correlated data.

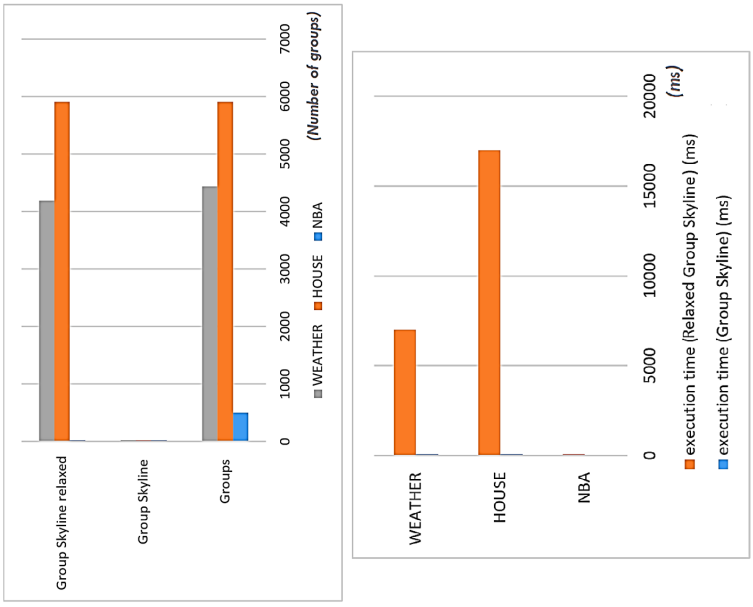


Fig. 6. Real data execution time and the number of relaxed skyline groups returned

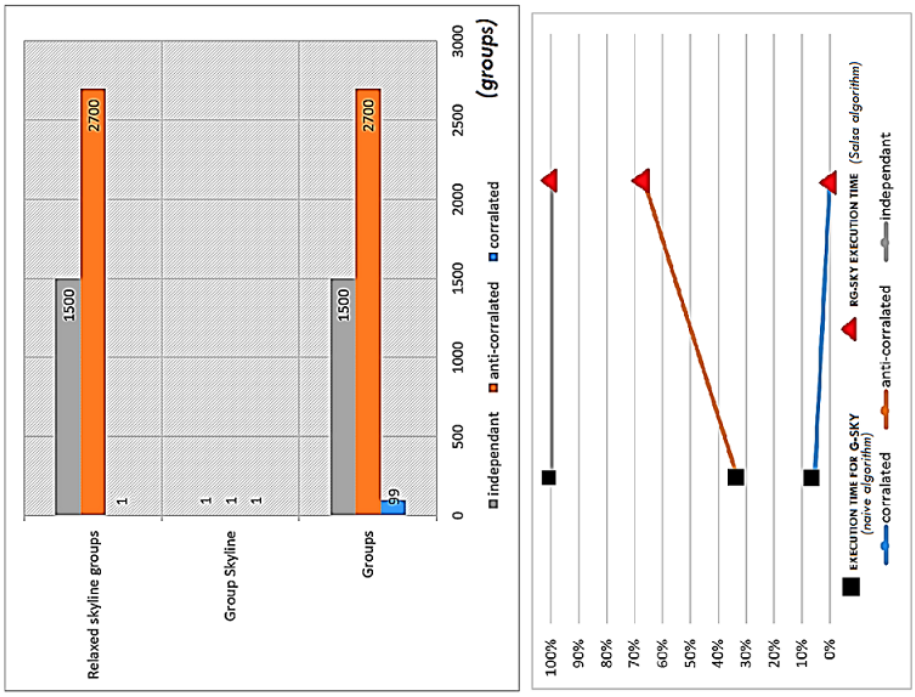


Fig. 7. Synthetic data execution time and the number of relaxed skyline groups in different data types

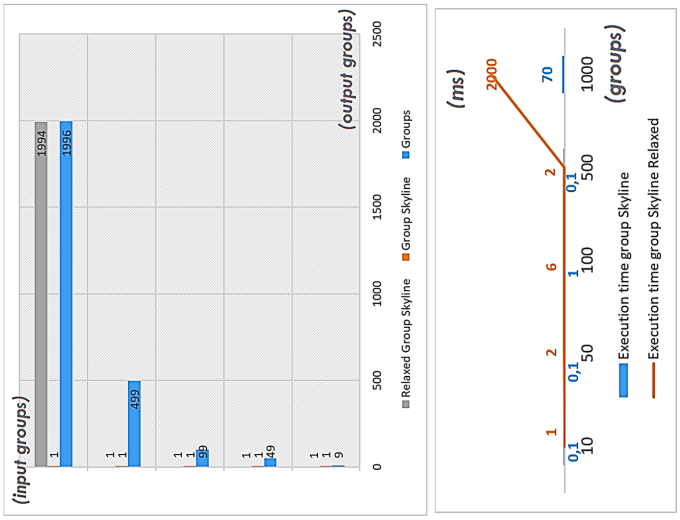


Fig. 8. Synthetic correlated data with different dataset size

– **Data size variation**
Figure 8 shows that the RG-SKY approach is very efficient. We obtain more skyline groups with almost similar time consumption when the input groups do not exceed 500 groups.

– **Group dimension variation**
In this part, we focus and vary the group dimensions (i.e., the parameter l) instead of the number of input groups (i.e. N).
The figure 9 shows that the RG-SKY approach time execution is acceptable when the number of elements (i.e. l) in the group does not exceed 4. For instance, for 100 tuples, if l=3 then the time execution for group generation: G-SKY computation and RG-SKY calculus are respectively 4754 (ms), 7 (ms), 19 (ms). While for l=5, we obtain 3612281 (ms), 3022 (ms), 242000 (ms) respectively.

Case 3: Generated Data

- **Different relaxation vector values**
For the last test, we generate groups using normalized data values in order to analyze the impact of the relaxation vector values on the RG-SKY approach. Due to data normalization, we use the same much preferred relation $MP_{\mathbb{G}_i}(\gamma 1,\gamma 2)$ for all the skyline attributes i.
Figure 10 shows two cases for the $MP_{\mathbb{G}_i}(\gamma 1,\gamma 2)$ relation:
  1. Case 1: $\gamma 1$ is fixed ($\gamma 1$=0) and $\gamma 2$ varies to increase the relaxation zone, the obtained result shows that the size of the relaxed group skyline increases when $\gamma 2$ is larger but the execution time remains acceptable.
  2. Case 2: $\gamma 1$ and $\gamma 2$ are both changing, the obtained result shows that the size of the relaxed group skyline increases similarly as the first case. One can observe that the execution time can be considered as reasonable w.r.t the numbers of the relaxed generated skyline groups.

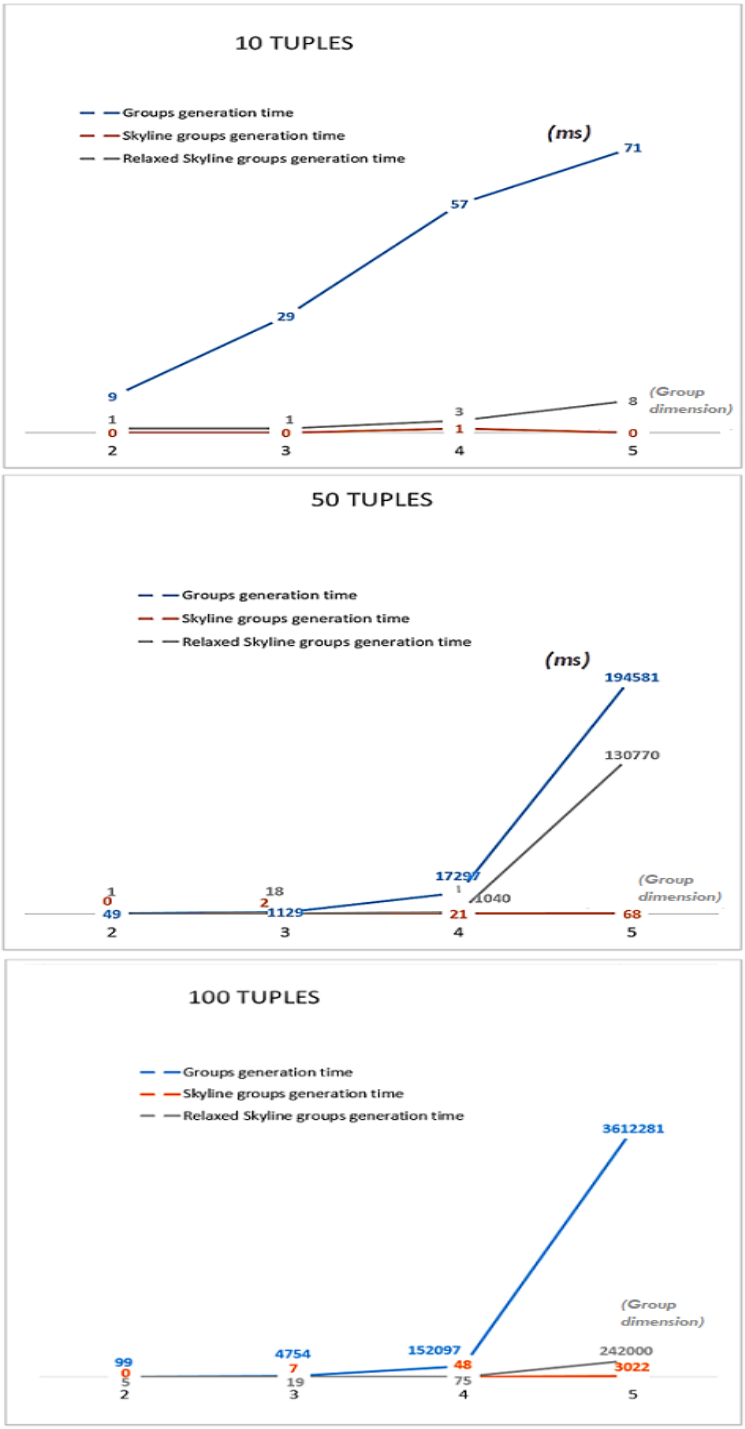


Fig. 9. Synthetic correlated data with a dynamic values of K-group (dimensions) and N-group (Number of groups)

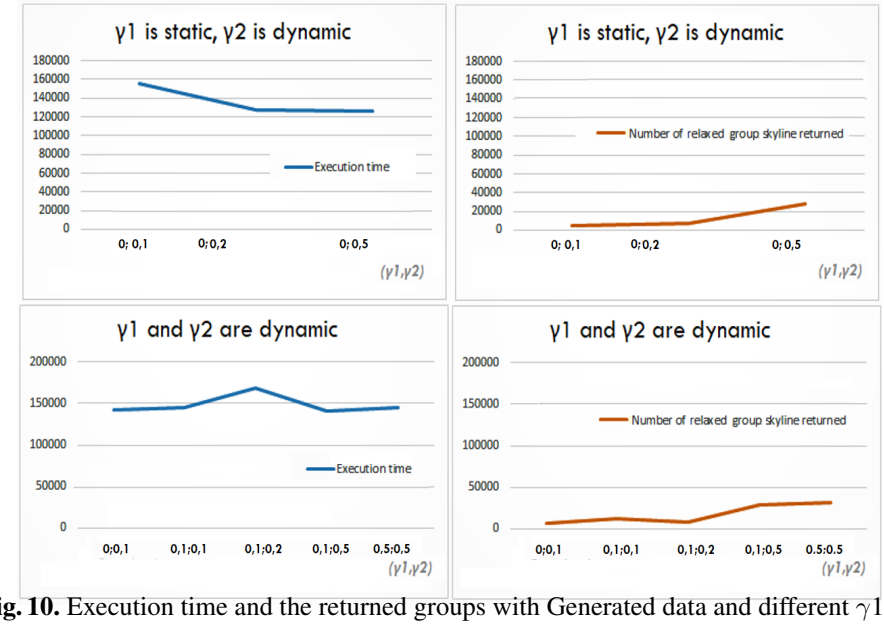


Fig. 10. Execution time and the returned groups with Generated data and different $\gamma 1$ and $\gamma 2$ values

7. Conclusion

In this paper, we addressed a new problem in the skyline community, that is, the problem of small group skylines. An approach for relaxing this kind of skyline, called RG-SKY, is discussed. It allows enlarging the group skyline at hand with interesting groups in a controlled way, and thus makes the decision easier. Moreover, to better meet the needs and expectations of the decision makers, the RG-SKY approach offers them the choice of the appropriate relaxation thanks to the different input parameters. The key concept of this approach is a particular fuzzy relation named much preferred whose semantics is user-defined. In addition, two algorithms to compute the relaxed group skyline are proposed, the first is a naive version of the RG-SKY method and the second is an optimized version using the Salsa algorithm which is used for the first time to extract groups instead of individual points in the relaxation context. The experimental study shows that the RG-SKY approach is a good alternative in terms of execution time, number of the relaxed skyline groups and user satisfaction.

As for future work, we plan to optimize the performance by taking the group generation part into consideration by eliminating groups using hierarchy methods based on the number of skyline points per group. We plan also to explore the parallel computation for the progressive relaxation generation of group skyline to optimize the time consumption of the approach, Finally, we try to generate the relaxation vector automatically taking into account the user attribute preferences.

References

1. Bartolini, I., Ciaccia, P., Patella, M.: Efficient sort-based skyline evaluation. ACM Trans. Database Syst. 33, 31:1–31:49 (2008)
2. Belkasmi, D., Hadjali, A., Azzoune, H.: On fuzzy approaches for enlarging skyline query results. Applied Soft Computing 74, 51–65 (2019)
3. Borzsony, S., Kossmann, D., Stocker, K.: The skyline operator. In: proc. 17th Inter. Conf. on data engineering. pp. 421–430. IEEE (2001)
4. Chester, S., Šidlauskas, D., Assent, I., Bøgh, K.S.: Scalable parallelization of skyline computation for multi-core processors. In: 2015 IEEE 31st Inter. Conf. on Data Engineering. pp. 1083–1094. IEEE (2015)
5. Chomicki, J., Godfrey, P., Gryz, J., Liang, D.: Skyline with presorting. In: 19th Inter. Conf. on Data Engineering. pp. 717–719 (March 2003)
6. Chung, Y.C., Su, I.F., Lee, C.: Efficient computation of combinatorial skyline queries. Information Systems 38(3), 369–387 (2013)
7. Cui, X., Dong, L.: An efficient algorithm to compute compositional skyline. In: IOP Conf. Series: Materials Science and Engineering. vol. 466, p. 012021. IOP Publishing (2018)
8. Dong, L., Liu, G., Cui, X., Li, T.: Finding group-based skyline over a data stream in the sensor network. Information 9(2), 33 (2018)
9. Dubois, D., Kerre, E., Mesiar, R., Prade, H.: Fuzzy interval analysis. In: Fundamentals of fuzzy sets, pp. 483–581. Springer (2000)
10. Godfrey, P., Shipley, R., Gryz, J.: Maximal vector computation in large data sets. In: proc. of the 31st Inter. Conf. on Very Large Data Bases. pp. 229–240. VLDB '05 (2005)
11. Goncalves, M., Tineo, L.: Fuzzy dominance skyline queries. In: Inter. Conf. on Database and Expert Systems Applications. pp. 469–478. Springer (2007)
12. Guo, X., Li, H., Wulamu, A., Xie, Y., Fu, Y.: Efficient processing of skyline group queries over a data stream. Tsinghua Science and Technology 21(1), 29–39 (2016)
13. Guo, X., Xiao, C., Ishikawa, Y.: Combination skyline queries. In: Transactions on Large-Scale Data-and Knowledge-Centered Systems VI, pp. 1–30. Springer (2012)
14. Im, H., Park, S.: Group skyline computation. Information Sciences 188, 151–169 (2012)
15. Jiang, T., Zhang, B., Lin, D., Gao, Y., Li, Q.: Incremental evaluation of top-k combinatorial metric skyline query. Knowledge-Based Systems 74, 89–105 (2015)
16. Kossmann, D., Ramsak, F., Rost, S.: Shooting stars in the sky: An online algorithm for skyline queries. In: proc. of the 28th Inter. Conf. on Very Large Data Bases. pp. 275–286. VLDB '02, VLDB Endowment (2002)
17. Li, K., Yang, Z., Xiao, G., Li, K., et al.: Progressive approaches for pareto optimal groups computation. IEEE Transactions on Knowledge and Data Engineering 31(3), 521–534 (2018)
18. Lin, M.Y., Lin, Y.L., Hsueh, S.C.: Discovering group skylines with constraints by early candidate pruning. In: Inter. Conf. on Advanced Data Mining and Appli. pp. 49–62. Springer (2017)
19. Liu, J., Xiong, L., Pei, J., Luo, J., Zhang, H.: Finding pareto optimal groups: Group-based skyline. proc. of the VLDB Endowment 8, 2086–2097 (2015)
20. Liu, J., Xiong, L., Zhang, Q., Pei, J., Luo, J.: Eclipse: Generalizing knn and skyline. arXiv preprint arXiv:1906.06314 (2019)
21. Nadouri, S., Hadjali, A., Sahnoun, Z.: Group skyline computation: An overview. In: proc. of the 36th Computer Workshop of Organizations and Information Systems and Business Intelligence Decision Making, Big Data and Data Science, INFORSID, France, May 28-31, 2018. (2018)
22. Nadouri, S., Ouhammou, Y., Sahnoun, Z., Hadjali, A.: Towards a multi-agent approach for distributed decision support systems. In: 2018 IEEE 27th Inter. Conf. on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE). pp. 72–77. IEEE (2018)
23. Nadouri, S., Sahnoun, Z., Hadjali, A.: Using g-skyline to improve decision-making. In: proc. of the 3rd Inter. Conf. on Advanced Aspects of Software Engineering, ICAASE 2018, Constantine, Algeria, December 1-2, 2018. pp. 141–150 (2018)

24. Papadias, D., Tao, Y., Fu, G., Seeger, B.: An optimal and progressive algorithm for skyline queries. In: ACM SIGMOD Inter. Conf. on Management of Data. pp. 467–478. SIGMOD '03, ACM, New York, NY, USA (2003)
25. Papadias, D., Tao, Y., Fu, G., Seeger, B.: An optimal and progressive algorithm for skyline queries. In: ACM SIGMOD Inter. Conf. on Management of Data. pp. 467–478. SIGMOD '03, ACM, New York, NY, USA (2003)
26. Papadias, D., Tao, Y., Fu, G., Seeger, B.: Progressive skyline computation in database systems. ACM Trans. Database Syst. 30(1), 41–82 (Mar 2005)
27. Papadias, D., Tao, Y., Fu, G., Seeger, B.: Progressive skyline computation in database systems. ACM Trans. Database Syst. 30(1), 41–82 (Mar 2005)
28. Perny, P., Roubens, M.: Fuzzy preference modeling. In: Fuzzy sets in decision analysis, operations research and statistics, pp. 3–30. Springer (1998)
29. Su, I.F., Chung, Y.C., Lee, C.: Top-k combinatorial skyline queries. In: Inter. Conf. on Database Systems for Advanced Applications. pp. 79–93. Springer (2010)
30. Tan, K.L., Eng, P.K., Ooi, B.C.: Efficient progressive skyline computation. In: 27th Inter. Conf. on Very Large Data Bases. pp. 301–310. VLDB '01, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001)
31. V, N.N.: Spatial skyline query algorithms. `https://viblo.asia/p/spatial-skyline-query-algorithms`, online-access June 2021
32. Yang, Z., Zhou, X., Mei, J., Zeng, Y., Xiao, G., Pan, G.: Identifying most preferential skyline product combinations. Inter. Journal of Pattern Recognition and AI 31(11) (2017)
33. Yang, Z., Zhou, X., Zeng, Y., Zeng, F., Zhou, Y.: Identifying most preferential skyline product combinations under price promotion. In: 2016 12th Inter. Conf. on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD). pp. 1824–1828. IEEE (2016)
34. Yu, W., Liu, J., Pei, J., Xiong, L., Chen, X., Qin, Z.: Efficient contour computation of group-based skyline. IEEE Transactions on Knowledge and Data Engineering (2019)
35. Yu, W., Qin, Z., Liu, J., Xiong, L., Chen, X., Zhang, H.: Fast algorithms for pareto optimal group-based skyline. In: proc. of the 2017 ACM on Conf. on Information and Knowledge Management. pp. 417–426. ACM (2017)
36. Zadeh, L.A.: Fuzzy sets. Information and control 8(3), 338–353 (1965)
37. Zhang, K., Gao, H., Han, X., Wang, J.: Finding k-dominant g-skyline groups on high dimensional data. IEEE Access 6, 58521–58531 (2018)
38. Zhang, N., Li, C., Hassan, N., Rajasekaran, S., Das, G.: On skyline groups. IEEE Transactions on Knowledge and Data Engineering 26, 942–956 (2014)
39. Zhou, X., Li, K., Yang, Z., Li, K.: Finding optimal skyline product combinations under price promotion. IEEE Transactions on Knowledge and Data Engineering 31(1), 138–151 (2018)
40. Zhu, H., Li, X., Liu, Q., Xu, Z.: Top-k dominating queries on skyline groups. IEEE Transactions on Knowledge and Data Engineering pp. 1–1 (2019)
41. Zhu, H., Li, X., Liu, Q., Zhu, H.: Computing skyline groups: an experimental evaluation. Tsinghua Science and Technology 24(2), 171–182 (April 2019)
42. Zhu, H., Zhu, P., Li, X., Liu, Q.: Computing skyline groups: An experimental evaluation. In: proc. of the ACM Turing 50th Celebration Conf. - China. pp. 48:1–48:6. ACM TUR-C '17, ACM, New York, NY, USA (2017)
43. Zhu, H., Zhu, P., Li, X., Liu, Q.: Top-k skyline groups queries. In: EDBT. pp. 442–445 (2017)
44. Zhu, H., Zhu, P., Li, X., Liu, Q., Xun, P.: Parallelization of group-based skyline computation for multi-core processors. Concurrency and Computation: Practice and Experience 29(18) (2017)

Sana Nadouri is a PhD student in Computer science at the university of Constantine 2, Constantine, Algeria and the National Engineering School for Mechanics and Aerotechnics (ISAE-ENSMA), Poitiers, France. She is a member of the Laboratory LIRE and the

laboratory LIAS. The areas of her scientific interest focus on Artificial intelligence, decision making and data extraction and optimization. The complete list of her publications is available in http://www.lias-lab.fr/members/sananadouri.

Allel Hadjali is currently Full Professor in Computer Science at the National Engineering School for Mechanics and Aerotechnics (ISAE-ENSMA), Poitiers, France. He is a member of the Data and Model Engineering research team of the Laboratory of Computer Science and Automatic Control for Systems (LIAS). His research interests are Massive Data Exploitation and Analysis, Extraction, Recommendation and Explainabiliy in Learning Machine Models. The complete list of his publications is available in http://www.lias-lab.fr/members/allelhadjali.

Zaidi Sahnoun got his Engineer degree from the university of Constantine, Algeria and the Master and PhD degrees from RPI, Troy N.Y, USA. He has held many scientific positions (head of the Computer Science Department, director of the LIRE laboratory and dean of the faculty of Computer Science and Information Technology at Constantine 2 University). Currently, Professor SAHNOUN is retired as a Full Professor and he is a member of the LIRE Laboratory. His main research interests are Software and Knowledge Engineering, Multi Agent Systems and Artificial Intelligence.

Received: October 20, 2021; Accepted: April 15, 2022.

Computer Science and Information Systems 19(2):913–934 https://doi.org/10.2298/CSIS211101009A

An Approach to Email Categorization and Response Generation

Sasa Arsovski[1], Muniru Idris Oladele[2], Adrian David Cheok[2], Velibor Premcevski[3] and Branko Markoski[3,*]

[1] Raffles University, Menara Kotaraya
Menara Kotaraya, Level 9, #09, 01, Jalan Trus, Bandar Johor Bahru,
80000 Johor Bahru, Johor
sasa.arsovski@gmail.com
[2] Imagineering Institute, Johor Malaysia
Anchor 5, Mall of Medini, 4, Lebuh Medini Utara,
79200 Nusajaya, Johor
idris@imagineeringinstitute.org
adrian@imagineeringinstitute.org
[3] University of Novi Sad, Technical Faculty "Mihajlo Pupin",
23000 Zrenjanin, Serbia
velibor.premcevski@tfzr.rs
markoni@uns.ac.rs

Abstract. The creation of automatic e-mail responder systems with human-quality responses is challenging due to the ambiguity of meanings and difficulty in response modeling. In this paper, we present the Personal Email Responder (PER); a novel system for email categorization and semi-automatic response generation. The key novelty presented in this paper is an approach to email categorization that distinguishes query and non-query email messages using Natural Language Processing (NLP) and Neural Network (NN) methods. The second novelty is the use of Artificial Intelligence Markup Language (AIML)-based chatbot for semiautomatic response creation. The proposed methodology was implemented as a prototype mobile application, which was then used to conduct an experiment. Email messages logs collected in the experimental phase are used to evaluate the proposed methodology and estimate the accuracy of the presented system for email categorization and semi-automatic response generation.

Keywords: email responder, deep learning, AIML, chatbot**.**

1. Introduction

Nowadays, there are many forms of digital communication such as Internet messaging, chat, social networking, etc. Despite all those communication forms, email remains the leading form of business communication [1]. With the huge increase in email overload, users find it very challenging to process and respond to all incoming messages [2].

* Corresponding author

According to [1], mobile email has shown rapid growth; currently, 65% of email users worldwide access their email via a mobile device. Authors in [3] stated that these days email overload has new forms. Users receive different types of emails that match multiple aspects of their life. Research findings in [3] indicate that email overload is present both at work and in private life. These findings suggest new opportunities for email overload research.

The main objective of the personal email responder system that will be presented in this paper is to help users to overcome email overload challenges and minimize user efforts in the email answering process. The first challenge toward building a PER is to identify which types of messages can be automatically or semi-automatically processed.

Email categorization and response generation systems deploy different approaches; however, they can be grouped into the three categories: 1) text retrieval approaches, 2) text categorization by machine learning and, 3) statistical text similarity calculation by matching of text patterns and templates [4].

After an initial study based on a log that contains 12,000 personal email messages, we categorize the email messages in to two main categories. The first category is the query email messages. Query email messages are messages that contain some question and require an answer. The second category is declarative and informative messages. Declarative and informative messages are messages that contain declarative sentences which do not require an answer, or the answer can be created automatically.

For the model construction, we will use Deep learning techniques, NLP tools and, AIML chatbot functionalities. Deep learning [5] is a set of algorithms in machine learning that attempt to learn on multiple levels, corresponding to various levels of abstraction. It typically uses artificial Neural Networks (NN) [6, 7].

The key novelty of this work is an approach that is applied for email message categorization. Also, in this paper, we will present novel system based on AIML functionalities for semiautomatic email response generation.

The rest of the paper is organized as follows: the second section contains related works; the third section provides the dataset analysis details and model architecture; results and discussions are presented in the fourth section. Section five contains our conclusion and future work.

2. Related Works

Automating distinctive features of email systems has a long history. One of the good examples is the work presented in [8]. Motivated by the need to lessen the burden of the large volume of emails of many users, the authors suggested a smart environment for email processing. The personal email assistant can prioritize, filter, refile the incoming emails based on classification, search through the emails, and make vacation responses. Although some of these tasks were already included in different email systems, the authors stated that none of them combined all those capabilities. As an email responder, proposed system creates appropriate routine responses to incoming messages. The compilation of these responses is based on the email subject, current schedule, and the content of the email. The personal email assistant proposed in [8] uses personal calendar, personal preferences, email context, and decides about response content.

According to the work in [4], automated email answering will be a text categorization task if all messages in one text category have the same standard answer. In [9], several machine learning algorithms were adopted. The algorithms included the k-NN, Naive Bayes, RIPPER, and SVM, and were implemented for generating automatic answers to 4490 technical support-related email messages. These messages were grouped into 47 categories; each category had no less than 30 messages. All experiments were carried out using tenfold cross validation. Support Vector Machine (SVM) shows the best performance, with accuracy (the share of correctly suggested standard answers) of 56% for a single answer, and for 78% of the email messages, the correct answer was among the top 5 suggested answers.

In [10], the sentence matching approach is adopted, however, this is a more challenging task than document categorization; it involves locating one or several questions in a message and selecting one or several standard answers to the questions. Query emails are analyzed against a database system which comprises several tag-question (FAQ) and standard answer pair. When the new e-mail arrives, the system analyzes the question in an email and provides an answer mapped to a predefined question template. When the system cannot find the appropriate question template, it must determine the similarity between the new questions and the existing question templates. In this case, the system will calculate the similarity between the concepts present in the sentences that are compared. Using the method defined in [11], to calculate overall similarity score between the two sentences, the system compares a new question with pre-defined set of questions templates. When it finds a most similar question template, the system takes an answer to the template question in response to a new question.

Kosseim et al. [12] use information extraction templates to (i) identify the query message the purpose, the sender, etc., (ii) extract named entities from the query message, (iii) extract relations between the main concepts, and (iv) capture domain specific relations. The next step is semantic validation. The system verifies whether the extracted data and the respective templates all together make any sense as an answer. The third step is an analysis of the obtained information and querying some external sources for new data to complete the answer. Finally, the system fills the answer template with the data and generates the answer text.

Text-pattern matching is another minority approach. Sneiders [4] has developed a technique that operates a set of manually crafted text patterns assigned to FAQs. A text pattern resembles a regular expression. It contains stems of words and their synonyms. It can match phrases, stand-alone words, and compound words. Each FAQ has one or several required text patterns (they must match a query) and one or several forbidden text patterns (they must not match the query). Experiments in two languages (Swedish and Latvian) and two domains (insurance and telecom) showed consistent results: if the system did retrieve an answer, the answer was correct in about 90% of the cases. The recall values were 68% and 76% in the respective language and domain.

Recently, there are a few works that propose state-of-the-art email auto-responder systems. Those approaches use Deep Learning techniques for creating automatic email responders.

In the [13] authors developed a knowledge-based Question-Answer system. The user has the possibility to ask questions to a Question-Answer system that will retrieve related questions from the mailing list and show the significant sentences that were

extracted the related questions and their answers. However, the system is limited only to "how" type questions. Google provides a system that analyzes the incoming email and generates a reply [9]. Then, the user can modify the reply, and based on this modification, the system will update the email analysis mechanism. The system consists of two parts: 1) filter and modeler - for performing language analysis and characterization of the email body, and 2) reply composer - for generating a response to incoming email, based on the library of phrases. This system is based on general concepts and does not reflect the persona of email owner.

In [14], the authors address two main prediction issues arising in email systems. The recommendation system can alert the user when the email needs a reply or attachment. This process is very important for facilitating email response management. Another work, presented in the patent from Google [15], describes a method of eliminating unwanted incoming emails. There is also research for using the NN as an engine for detecting spam messages [16]. However, all these works were directed to automate specific functionalities of various email systems, and none of them is a fully automatic system which can express the personality of the email owner. A similar approach is presented in Mahendra [17]. The author suggested a personified email response generator using a deep neural network (DNN). This system, based on the previous email conversations of the owner, can generate a new message with the same style even for the emails that belong to the invitation or meeting category. The method is based on the Seq2Seq model using Long-Short Term Memory and Gated

Recurrent Units that help to obtain satisfactory results for long sentences. The final purpose of the author is to generate a response that will not require any editing afterward. Hence, right now, the system suggested in this work is the personal email assistant that is not yet able to reply automatically to all incoming emails.

All these works are in a various way starting from question answering systems until semi-automatic replying systems. In addition, almost all related works discuss email answering systems in the closed domain which are based on various technologies and apparatus. Our system extends Busemann work [18] into the next stage.

3. Data Analysis and Model Architecture

In this section, we will present analysis details of the email messages dataset and the architecture of our model for email categorization and response generation.

3.1. Email Dataset Analysis and Message Categorization

For the experiment, we use 12,000 randomly chosen personal Gmail messages. Our approach was based on the following hypothesis:

H_1: If an email body text contains a question, then the user needs to answer to this email - the model generates an automatic or semi-automatic answer. If email body text contains statements without questions, the system will present the user the email.

Our approach to message categorization is similar with the work described in Busemann et al. [18]. Optional input for machine learning was presence or absence of linguistic constructions frequent in questions and problem descriptions:

Table 1. Clause level tags

Tag	Description
S	Simple declarative clause, i.e. one that is not introduced by a (possibly empty) subordinating conjunction or a wh-word and that does not exhibit subject-verb inversion
SBAR	Clause introduced by a (possibly empty) subordinating conjunction
SBARQ	Direct question introduced by a wh-word or a wh-phrase. Indirect questions and relative clauses should be bracketed as SBAR, not SBARQ.
SINV	The inverted declarative sentence, i.e. one in which the subject follows the tensed verb or modal.
SQ	Inverted yes/no question, or main clause of a wh-question, following the wh-phrase in SBARQ.

Table 2. E-mail data analysis

Tag	Number of Emails in dataset
SQ	3169
SQ+SBRQ	8326
All messages	12000

Negation at the sentence or phrase level, yes/no and who-when-what-why-where-which-questions. We analyzed email messages dataset using a Penn Tree Bank clause level tags (Table 1) defined in [19]. For analyzing an email dataset, we use Stanford parser Java library [20] and query the dataset against SQ and SBARQ tags. The analysis results are shown in Table 2.

As shown in Table 2, 69.38 % of email messages in the analyzed dataset contain questions and 26.63 % are SQ questions. Based on those results, we decide to group messages into two main categories. The first category is the query email messages. Query email messages are messages that contain one or more question and require answer. The second category is declarative and informative messages. Declarative and informative email messages are messages that contain declarative sentences which do not require answers or answers can be created automatically. Message categorization of the analyzed dataset is shown in Fig.1.

As shown in Fig.1, In the query email messages category, three types of email messages are identified: Polar question email messages (yes/no questions), Factual question email messages and the combination of those two types.

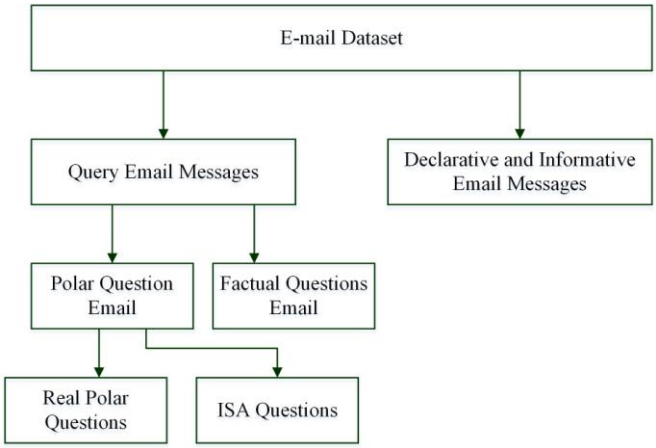


Fig. 1. Email messages categorization

A polar question email messages are messages that contain one or more polar questions. Polar question is a question which has only two possible responses: a" yes" which is an affirmative response or a "no" which is a negative) response [21]. The example of an email message with multiple polar questions is illustrated below:

Hi John,

Thanks again for the meeting, it was really lovely to meet you. Now, hopefully, you've got a bit of insight on what we do too after Mary presentation. Are you still interested in collaboration, and can I place you as a collaborator on my application for the Academy? Thank you!

Best wishes,

Factual question messages are messages that contain one or more Wh-questions. Wh-questions are questions which start with a question-asking word, either a Wh-word (what, when, where, which, who, whose, why) or questions with the word how [22]. The example of Factual question email message type is shown below:

Dear John,

To how many people would be talking about the Conference?

How much is the fee you are offering to Anna? Many thanks.

After initial analysis of the email messages using the Stanford Parser [20], we identified that in the case of Indirect Speech Act (ISA), especially when Factual (Wh) questions are asked in the form of a Polar question, the parser classifies those messages as Polar question emails. According to Yule [23], ISA is defined as an act when the speaker does not explicitly state the intended meaning of the utterance. It is the hearer's task to analyze the utterance to understand its meaning. An example of an ISA email message:

Dear Prof,

Please, could you send to me date and time of your arrival? Thank you!

Best wishes

3.2. Model Architecture

The main objective of our model is to help users to overcome email overload challenges and minimize user efforts in the email answering process. Our model needs to enable automatic and semi-automatic response generation to the query email messages category. The first step in our approach is to categorize email message. Using the Stanford parser [20], we grouped email body text sentences into main categories (Query email messages, and Declarative and informative email messages), using the methodology defined in the previous paragraph. Depending on the email message category, our system decides how to create email response. Fig.2 depicts workflow of the proposed system for answer generation.

As shown in Fig.2, our system contains two main modules. The first module gathers all unread email messages from a user's Gmail account and parses those messages using Stanford NLP. After parsing email message body texts, the system does the extraction of questions from the email message body text and passes questions and email text to the second module. The second module is a query module. In this module, we will introduce *three-query layers:*

The first layer detects if the email message body text consists of SQ or SBARQ question types. If not, the layer will present the user with an email message and finish processing. If email body text consists of an SQ or SBARQ type of questions layer will pass all information to the second query layer.

The second query layer detects SQ question types. If the message does not have an SQ question, the system will create a custom response template for an SBARQ question type and ask the user to fill the gaps in the recommended response template (template generation will be explained in the following text). If an SQ question type is detected, all questions are passed to the third query layer.

In the third query layer, the system will check if the passed SQ question is a real SQ question or an ISA type of question. For ISA type question detection, we will use a Convolutional Neural Network (CNN) for multi label text categorization. If an ISA question type is detected, the system will present the user with a question and wait for a user action (yes/no) answer. After the user action, the system will pass the question and user action indicator to the AIML chatbot. Depending on the value of the user action indicator, the chatbot will generate *yes* or *no* type of answer and present the user with an answer. The system allows the user to edit the generated answer. If query layer detects that it is a real SQ type of question, the question will be passed directly to the AIML chatbot for automatic answer generation.

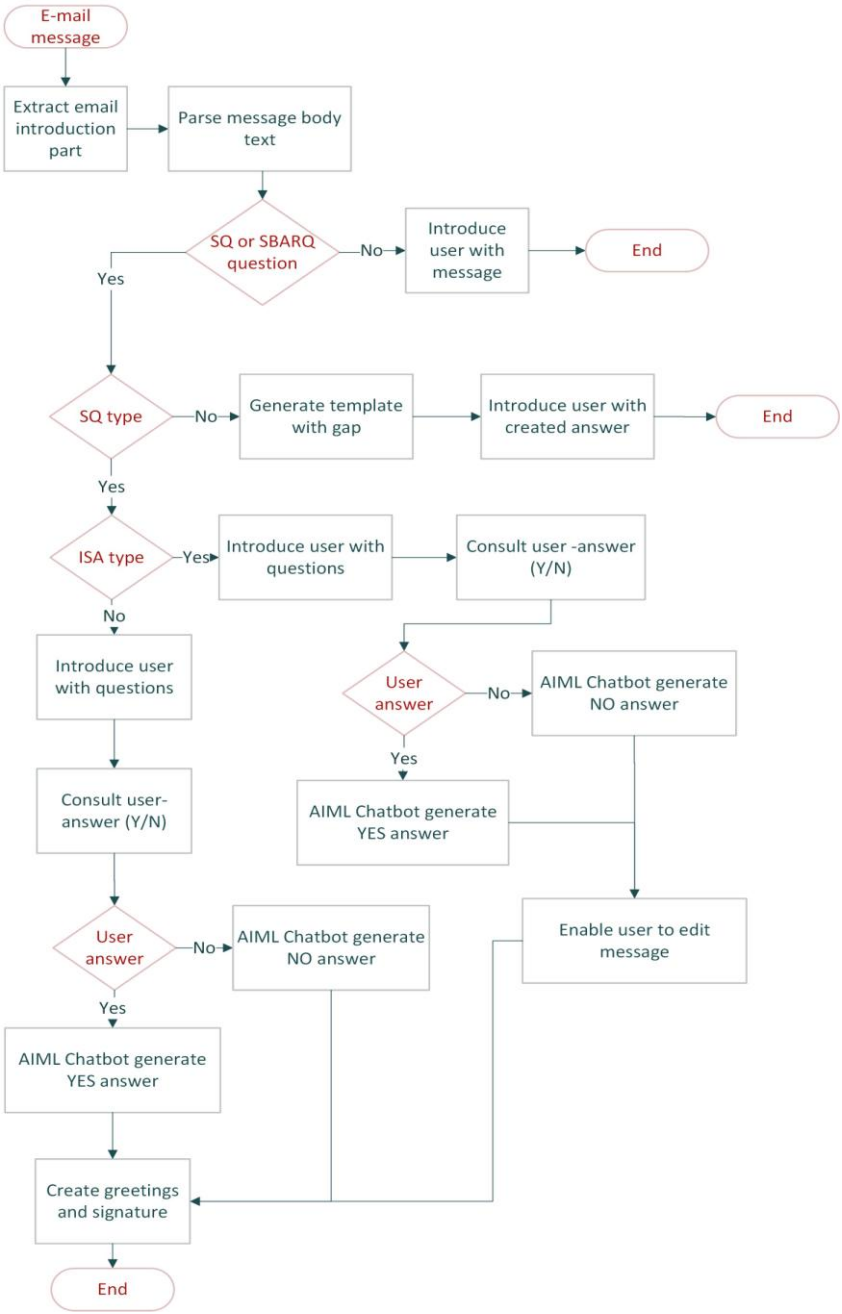


Fig. 2. Email answer generation – Workflow

3.3. Template Generation for the SBARQ Question Type

For the template generation, we proposed the use of LSTM seq2seq NN model as is described in [2]. LSTM is a special kind of RNN. Seq2seq is an NN model that uses one LSTM to read the input sequence encoder, for every time step and the result is a vector. It then uses another LSTM to extract the output sequence from that vector decoder (Fig.3) [24]. The goal of LSTM is to estimate the conditional probability $p(y_1,...,y_{T0}|x_1,...,x_T)$, where $(x_1,...,x_T)$ is an input sequence (words) and $(y_1,...,y_{T0})$ is its corresponding output sequence (words) whose length T^0 may differ from T. The LSTM computes this conditional probability by first obtaining the fixed dimensional representation v of the input sequence $(x_1,...,x_T)$ given by the last hidden state of the LSTM, and then computing the probability of $y_1,...,y_{T0}$ with a standard LSTM-LM formulation whose initial hidden state is set to the representation v of $x_1,...,x_T$:

$$p(y_1,...,y_{T0}|x_1,...,x_T) = \prod_{t=1}^{T_0} p(y_t|v,y_1,...,y_{t-1}) \qquad (1)$$

In this equation, each $p(y_t|v,y_1,...,y_{t-1})$ distribution is represented with a SoftMax over all the words in the vocabulary. The overall scheme is outlined in Fig.3, where the shown LSTM computes the representation of $A,B,C<EOS>$ (sequences of the input data) and then uses this representation to compute

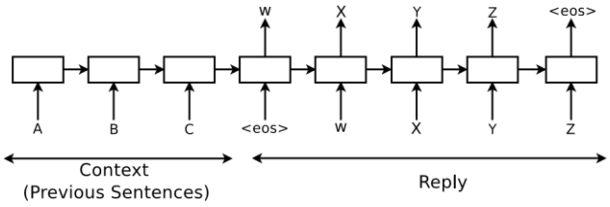


Fig. 3. seq2seq model [24]

the probability of $W, X, Y, Z < EOS >$ (sequences of the output data).

When the system passes the SBARQ question to the NN model, it will create an automatic answer. Using a Stanford NLP named entity recognition, the system will recognize and remove all named entities from the generated answer and present the user with a template. The system will enable to user to fill the gaps made in the named entities extraction process. The proposed PER functionality is not implemented at the time of writing this paper since we were unable to collect enough personal messages to train the proposed NN model.

3.4. ISA Question Type Categorization

For ISA question detection, we use CNN model for multi-label text categorization. Convolutional Neural Networks (CNN) are used for multi-label text classification tasks. CNN is a category of NN that has been proven very effective in areas such as image recognition and classification [25]. CNNs have been successful in identifying faces,

objects and traffic signs apart from powering vision in robots and self-driving cars [26]. CNNs are a type of feedforward artificial NNs in which the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex [25, 27]. CNNs exploit spatially local correlation by enforcing a local connectivity pattern between neurons of adjacent layers as shown in the Fig.4.

A feature map is obtained by repeated application of a function across subregions of the entire image, in other words, by convolution of the input image with a linear filter, adding a bias term and then, applying a nonlinear function. If we denote the k-th feature map at a given layer as h^k, whose filters are determined by the weights W^k and bias b_k, then the feature map h^k is obtained as follows (for *tanh* non-linearity's):

$$h_{ij}^k = (W^k * x)_{ij} + b_k \quad (2)$$

For training the NN model we use a dataset of 1000 human detected ISA type questions. We train a model using batched stochastic gradient descent (SGD), which is a standard choice for CNNs. SGD is a stochastic approximation of the gradient

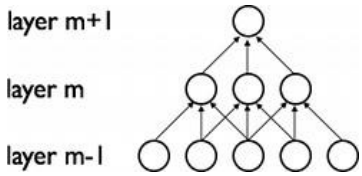


Fig. 4. A simple illustration of the CNN

descent optimization method for minimizing an objective function that is written as a sum of differentiable functions [28]. We randomly select batches of data from the training set (stochastic gradient descent). Tuning the batch size is one of the aspects of getting training right - if a batch size is too small, then there will be a lot of variance within a batch. If a batch size too large, the model will run out of memory, or training will progress too slow. Epoch is defined as one pass through the training data. There are multiple batches in each epoch. The learning rates (LR) is directly connected with the batch size. In CNN, every network layer acts as a detection filter for the presence of specific features or patterns present in the original data [5].

We define training parameters as suggested in Sneiders [4]. The model has the following parameters: num. epochs= 1, batch size=37, num. filters=32, filter sizes= 3,4,5, embedding dimension=200 and drop out = 0.5. By running prediction on the test dataset, we achieved 84.02% accuracy in ISA question type classification. Examples of the email categorization will be shown in the fourth section of this paper.

3.5. AIML-based Chatbot for Semi-Automatic Response Creation

The key novelty presented in this paper is semi-automatic email response generation powered by the Artificial Intelligence Markup Language (AIML). AIML is an XML compliant language for authoring Questioning-Answering agent [29]. We leveraged on an existing implementation through Program AB; a Java programming language interpreter for AIML [30].

Functionally, an AIML category defines a question-answer pair which is represented with the category tags denoted as < category >< /category >. In the category tags are the pattern, which stores the expected input from the users and template tags, which contains the automatic response for the respective input. A simple AIML category is illustrated in Listing 1.

Listing 1- A simple AIML category

```
<category >
<pattern >Hello * <pattern >
<template >I am fine </ template >
</category >
```

According to the presented categorization of the emails, the email dataset comprises the polar and non-polar questions [31]; however, our focus was on the polar questions. Polar questions are generally called yes/no questions and they are usually defined by a "tensed auxiliary clause-initially" [32]. A tensed auxiliary clause-initially is formed by transforming an indicative-mood statement to an interrogative-mood statement, generally by positional transformation of the subject-(auxiliary verb) order in the indicative-mood statement [33], for instance, "I can" to "Can you ". Auxiliary verbs (AV) are also called the "helping verbs". Though they do not introduce new semantic content in a clause, they express tenses, moods, voices or modality. According to Kies [34], AV can be grouped into Table 3:

Table 3. An illustration of the mood in sentences

	Indicative-mood state	Interrogative-mood statement
Word order	{*S*| *Av*| *Mv*}	{*Av*| *S*| *Mv*}
Example	I can bring the letter	Can you bring the letter
	I am ready to sign the papers on his behalf	Are you ready to sign the papers on his behalf

**S, Av and Mv* are the subject, the auxiliary verb and the main verb of the clause respectively

the modal auxiliaries (can, could, should, may, would), perfect auxiliaries (has, have), progressive auxiliaries (is, are), passive auxiliary (were) and auxiliary support (do). An indicative-mood statement expresses facts in an honest, direct, relevant way while an interrogative-mood statement depicts a desire for information. Table 3. illustrates two examples of such transformations; for instance, by changing the order of the words, I (subject) and can (modal auxiliary verb) in the sentence "I can come with the letter", the polar question, "Can I come with the letter" is formed. The basic structure for PQ is illustrated below:

PQ = {Av + S + P}, where PQ is Polar Question, Av is

Auxiliary verb, S is the Subject and P, the Predicate (this includes the main verb and all other details that describe what is going on in the question).

Generally, responses to PQ are type-conforming; they usually conform with the constraints set by the grammatical structure of the question [35]. In other words, the responses usually agree with the function of the PQ, since a confirmation (positive or negative) rather than a new piece of information is provided Selting and Couper-Kuhlen [35], for instance, the question "Can you bring the letter" only requires a confirmatory

answer (yes or no). Based on this premise and manual examination of our dataset, we crafted syntactic structures or frames to represent patterns that capture the variations in the dataset; of which the auxiliary verb, the subject, root (main) verb and pronouns were key components. However, special attention was given to the pronouns; some assume different forms in question-and-answer mode. For instance, a comprehensive answer to the PQ "Can you give me the letter?" is "Yes, I can give you the letter"; the subjective pronoun (you) changes to a quote while the objective pronoun (me) becomes "you". Table 3. shows our crafted syntactic frames and examples of e-mails that fit match with them.

Furthermore, as mentioned earlier the answers to polar questions are type-conforming, the successful crafting of syntactic frames for the email dataset paved the way for easy response generation using an *AIML* parser.

The syntactic frames we crafted were logically transformed into AIML categories and implemented using the Java powered AIML interpreter (Program AB). The interpreter was run as a web service and four different implementations were deployed. They are:

Yes: used to provide a positive response

No: used to provide negative response

Short Yes: for short positive response

Typically, the AIML interpreter comprises the knowledge base (a collection of categories) and an input-response pattern matching algorithm. When the AIML interpreter gets an input (PQ), it searches through its knowledge base to determine which *category* has a *pattern* that the input exhibits. If a match is found, the value of the respective *template* tag will be fetched and sent as the output. Special tags (set, map, star) and wildcard characters (ˆ, *) are generally used as variable placeholders in the *pattern* "or/and" *template* tags to capture differing possibilities as explained in [30]. Table 5 shows an example of questions and respective responses generated by our AIML-based chatbot.

Table 4. Syntactic frames for the email dataset along with examples

Syntactic Frames	Original E-mail Sample	Framed E-mail Sample
(SF) + AV + {NP | PN } + (SF) + RVB + (SF) + (PN) + (SF) + (PN) + (SF) One pronoun or noun phrase before the root verb Zero or more pronouns after root verb	Adrian and Lonce, would you be in town on 4 May for this? Dear Sir, can I get another 3 days for leave? Are you free at the night? Dear Sir, will Karthik be subjected to the same grading system as an intern for payment? Has the international jury set up yet? As virtual experiences of extreme thrills become more ever-present, will we be satisfied with mere simulating danger? Hi Adrian, can you send to us the detailed proposal for QAFU pls? Did you book your ticket to Sri Lanka? Do you think they will be available? Can you kindly check this? Dear Sir, would you like our bio interns to start the DNA project immediately after their exams or when term starts. Will websites and virtual spaces continue to make it easier for us to connect with like-minded people and create strong, active, supportive communities.	{Adrian and Lonce}SF , + {would}AV + {you}PN + {be}RVB + {in town on 4 May for this}SF {Dear Sir ,}SF + {Can}AV + {I}PN + {get}RVB + {another 3 days for leave}SF {Are}AV + {you}PN + {free}RVB + {at the night}SF {Dear Sir}SF + {Will}AV + {Karthik}NP +{be subjected}RVB + {to the same grading system as an intern for payment}SF {Has}AV + {the international jury}NP + {set up}RVB + {yet}SF {As virtual experiences of extreme thrills become more ever-present}SF + {will}AV + {we}PN + {be satisfied}RVB + {with mere simulating danger}SF {Hi Adrian,}SF + {Can}AV + {you}PN + {send}RVB + {to}SF + {us} PN + {the detailed proposal for QAFU} + {pls}SF {Did}AV + {you}PN + {book}RVB + {your}PN + {ticket to Sri Lanka?}SF {Do}AV + {you}PN + {think}RVB + {they}PN + {will be available}SF {Can}AV + {you}PN+{kindly}SF + {check}RVB + {this}PN {Dear Sir, }SF + {Would}AV + {you}PN + {like}RVB + {our}PN + {bio interns to start the DNA project immediately after}SF + {their}PN + {exams or when term starts}SF {will}AV + {websites and virtual spaces}NP + {continue to make}RVB + {it}PN + {easier for}SF + {us}PN + {to connect with like-minded people and create strong, active, supportive communities}SF
(SF) + AV + {NP | PN} + SF + PN + (SF) + RVB + (SF) + PN + (SF) + (PN) + (SF) A noun phrase and a pronoun or two pronouns before the root verb Zero or more than one pronoun after the root verb	Could you and your laboratory accept my students Is it possible for me to arrange a meeting sometime this week, for all of you to discuss this exploratory trip together pls? Could you also let me know by return whether she has any allergies or medical issues?	{Could}AV + {you}PN + {and}SF + {your}PN + {laboratory}SF + {accept}RVB + {my}PN + {students}SF {Is}AV + {it}PN + {possible for}SF + {me}PN + {to arrange}RVB + {a meeting sometime this week, for all of}SF + {you}PN + {to discuss this exploratory trip together pls}SF {Could}AV + {you}PN + {also}SF + {let}RVB + {me}PN + {know by return whether}SF + {she}PN + {has any allergies or medical issues}SF

SF is a sentence fragment; Av is an auxiliary verb; NP is a noun Phrase; PN is Pronoun, RVB is the root verb in the question and | implies "OR". Note that an item enclosed within a bracket can either be empty or not. *Short No*: for short negative response.

Table 5. Sample questions and their corresponding AIML-generated responses

Question	Yes	No	Short Yes	Short No
Would you be able to come to Portugal	Yes I would be able to come to Portugal	No I would not be able to come to Portugal	Yes I would	No I would not
Is it possible to stick a screen somewhere for them for that purpose	Yes it is possible to stick a screen somewhere for them for that purpose	No it is not possible to stick a screen somewhere for them for that purpose	Yes it is possible	No it is not possible.
As one of the organizers, would you be able to send me summary of the reports	Yes I would be able to send you summary of the reports	No I would not be able to send you summary of the reports	Yes I would	No I would not
Has the international jury set up yet	Yes the international jury has set up yet	No the international jury has not set up yet	Yes the international jury has	No the international jury has not.
Dear Sir, would you like our bio interns to start the DNA project immediately after their exams or when term starts	Yes I would like your bio interns to start the DNA project immediately after their exams or when term starts	No I would not like your bio interns to start the DNA project immediately after their exams or when term starts	Yes I would	No I would not
Will websites and virtual spaces continue to make it easier for us to connect with like-minded people and create strong, active, supportive communities	Yes websites and virtual spaces will continue to make it easier for you to connect with like-minded people and create strong, active, supportive communities	No websites and virtual spaces will not continue to make it easier for you to connect with likeminded people and create strong, active, supportive communities	Yes websites and virtual spaces will	No websites and virtual spaces will not
Could you also let me know by return whether she has any allergies or medical issues?	Yes I could let you know by return whether she has any allergies or medical issues?	No I could not let you know by return whether she has any allergies or medical issues?	Yes I could	No I could not
Dear Sir, Will Karthik be subjected to the same grading system as an intern for payment?	Yes Karthik will be subjected to the same grading system as an intern for payment	No Karthik will not be subjected to the same grading system as an intern for payment	Yes Karthik will	No Karthik will not
Are you available on the morning of Thursday 12th June for the filming?	Yes I am available on the morning of Thursday 12th June for the filming	No I am not available on the morning of Thursday 12th June for the filming	Yes I am	No I am not
Am I right in assuming this is wrong and the correct line-up should be Adrian and Buchanan?	Yes you are right in assuming this is wrong and the correct line-up should be Adrian and Buchanan	No you are not right in assuming this is wrong and the correct line-up should be Adrian and Buchanan	Yes you are	No you are not

4. Implementation and Evaluation

4.1. Model Implementation

The presented model of the system is implemented as a mobile application. The application consists of two modules. The first is a backend module; a web service that queries a user's Gmail inbox and gathers all unread email messages. Then, it processes the gathered messages (parses email message text, recognize PQ, extract recognized questions) and writes results in the database. This process is repeated every two minutes.

The frontend part of our system is a mobile application. This application reads messages processed by the backend module and presents the user with the results. As shown in Fig.5, the user can swipe right to answer *yes* to a request or email question, or swipe left to *decline (No)* to a request or email question. Upon capture of the user action by the mobile application, the question is passed to the respective AIML chatbot and receives an answer.

4.2. Model Evaluation

In our experiment, a group of 40-people installed a mobile application and registered their Gmail account into our system. We analyzed 424 email messages and system generated answers to evaluate model accuracy. Categorization of the obtained dataset gives us results that are shown in Table 6.

Table 6. E-mail categories and number of questions

Tag	Number of categorized emails	Number of questions in categorized emails
SQ	94	234
SQ+SBRQ	95	342
All messages	424	

Analyzing results of the email categorization that are shown in Table 6, we can conclude that our "real life dataset" is similar to the initially analyzed dataset and have 22% of question emails. Also, we can conclude that most of the email messages consist of more than one question. Responses generated by our system are shown in Table 7. Our system can answer semiautomatically to 63.3% of the tested polar questions according to the human evaluation (1 is an acceptable answer 0 is an unacceptable answer). Further, proposed system recognizes ISA and enables the user to edit a generated message and significantly decrease the time of response creation. In 13 % of the cases, answers that are generated by AIML chatbot can be used with a good accuracy

We further carried out a human evaluation of the PER responses. The human evaluator analyzed 94 emails and provided responses to them. These responses were compared with responses generated by the PER using the BLEU score [36]. The BLEU score algorithm compares the N-grams of two text fragments and counts the number of matches, the similarity score of these texts is a function of the number of matches [37]. The BLEU score ranges from 0 to 1; values between 0.5 and 1 reveals a high similarity. We categorized BLEU score results in to 10 categories as is shown in Table 8. Fig.6 shows a plot of the BLEU score categories and number of emails in the respective category BLEU scores of at least 40% (0.4) while 46 have scores of at least 50% (0.5). This implies that the responses from the PER and the human evaluator have high level of similarity.

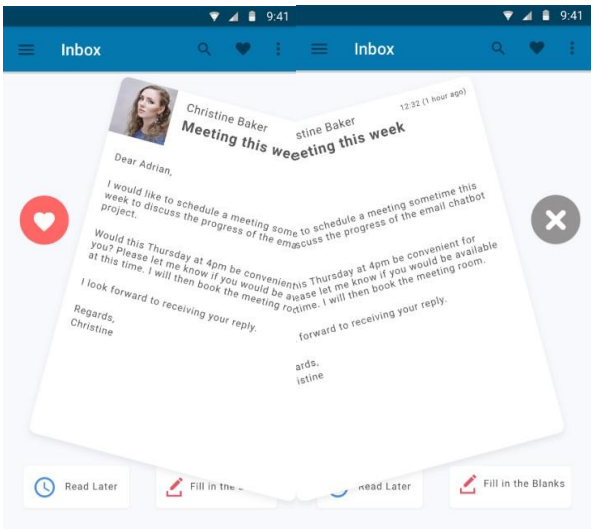


Fig. 5. Screen shot of the mobile app with the Yes/No functionality

5. Conclusion

In this paper, we present a novel system for email categorization and semi-automatic response generation, Automatic Personal Email Responder (PER). The main objective of the Automatic Personal Email Responder system which is presented in this paper, is to help users overcome email overload challenges and minimize user efforts in the email answering process. Our approach was based on the following hypothesis:

H_1: If an email body text contains a question then the user needs to answer to this email - the model generates an automatic or semi-automatic answer. If email body text contains statements without questions, the system will present the user the email.

Follow the hypothesis H_1, the key novelties that are presented in this paper are:

Email categorization approach that distinguishes query and non-query email messages using Natural Language Processing (NLP) and Neural Network (NN) methods,

Implementation of the Artificial Intelligence Markup Language (AIML) - based chatbot for semi-automatic response creation,

Application of template structured database for question answering (AIML) in email answering systems.

Table 7. System responses to questions in e-mails

Processed email text	Extracted polar question	Categorized question		Answers generated by the AIML chatbot
		SQ	ISA	
Lord Vader, Did the dark side reach a conclusion about my visa application?? People keep asking me if I am coming	Did the dark side reach a conclusion about my visa application??	Did the dark side reach a conclusion about my visa application??	-	Yes the dark side did reach a conclusion about your visa application
Dear ***** organizing committee I am ***** in ***** University, Japan. I will present in ***** poster session (submission no.70). Then, I have two questions about facilities and equipment in poster session. Is there a *desk* to put PC and our prototype devices in presentation space? I want to use a desk to explain our system actually. Best regards.	Is there a * desk * to put PC and our prototype devices in presentation space	Is there a * desk * to put PC and our prototype devices in presentation space	-	Yes ther is a desk to put PC and your prototyped devices in presentation space
Interested in your product. Could you send me prices and how to get?	Could you send me prices and how to get?	-	Could you send me prices and how to get?	Yes I could send you prices and how to get
Hi, I thought it would be great for us to sit down and chat. I am free Tuesday and Wednesday. Can you do either of those days?	Can you do either of those days?	Can you do either of those days?	-	Yes I can do either of those days
Why are we sending this? We take security very seriously and we want to keep you in the loop on important actions in your account. We were unable to determine whether you have used this browser or device with your account before. This can happen when you sign in for the first time on a new computer, phone or browser, when you use your browser's incognito or private browsing mode or clear your cookies, or when somebody else is accessing your account. The Google Accounts team	Why are we sending this	-	-	-

Dear *****, How are you today. Could you please confirm if you will be available for our meeting tomorrow	Could you please confirm if you will be available for our meeting tomorrow.	Could you please confirm if you will be available for our meeting tomorrow.	-	Yes I could confirm if I will be available for your meeting tomorrow
Dear Tita, Any progress about our order? Have you receive the payment?	-	-	-	
Hi Rasyidah, Could I follow up on this order. We haven't receive payment for this order.	Could I follow up on this order.	Could I follow up on this order.	-	Yes you could follow up on this order
Dearest Norbert Sorry for all the inconvenience, my staff are working like crazy to make this happen. Would you be available around mid to end of November? We will make it a very important Academic Advisor and Examiner meeting. So your trip will be very serious.	Would you be available around mid to end of November?	Would you be available around mid to end of November?	-	Yes I would be available around mid to end of November
Dear Azrina, Hope all is well. Is there a date for the board meeting in September? Thank you, all the best,	Is there a date for the board meeting in September	Is there a date for the board meeting in September	-	Yes there is a date for the board meeting in September
Dear Adrian, dear Azrina, I hope everything is fine with you. Can you please let me know if you agree with the dates of my visit that I suggested last Friday? They are within the time slot (second half of November) you proposed before. We can talk about the details of the agenda schedule later if you don't have time now. Looking forward to your confirmation. Thank you and best regards -	Can you please let me know if you agree with the dates of my visit that I suggested last Friday?	Can you please let me know if you agree with the dates of my visit that I suggested last Friday?	-	Yes I can let you know if I agree with the dates of your visit that you suggested last Friday

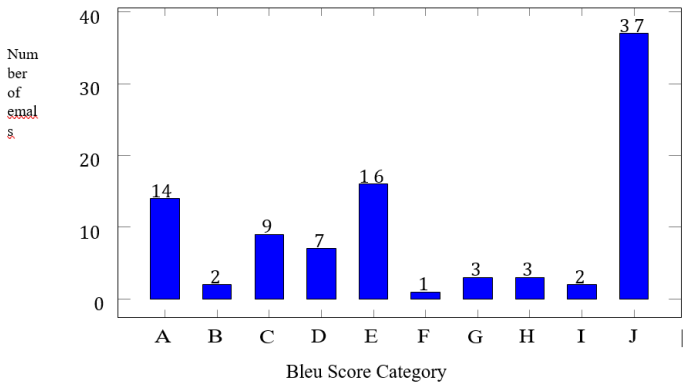


Fig. 6. Summary of BLEU score evaluation of 94 emails

Table 8. BLEU score Categorization

$BlueScore$	$Category.Notation$
$0 - 0.1$	A
$> 0.1\ and <= 0.2$	B
$> 0.2\ and <= 0.3$	C
$> 0.3\ and <= 0.4$	D
$> 0.4\ and <= 0.5$	E
$> 0.5\ and <= 0.6$	F
$> 0.6\ and <= 0.7$	G
$> 0.7\ and <= 0.8$	H
$> 0.8\ and <= 0.9$	I
> 0.9	J

Model evaluation results show that the proposed system can answer to 63.3% question emails correctly from the testing dataset. To increase accuracy of the presented system, we need to create a new NN model for ISA question classification using a bigger dataset. In future works, we will extend the functionality of our system to provide automatic responses to factual questions. We propose implementation of an LSTM seq2seq NN model to create answer templates as responses to the factual questions. Also, we will extend the scope of our system to analyze declarative and informative email messages.

References

1. "Email Statistics Report, 2016-2020," http://www. radicati.com/wp/wp-content/uploads/2016/01/Email Statistics Report 2016-2020 Executive Summary.pdf, 2016.
2. A. Kannan, K. Kurach, S. Ravi, T. Kaufmann,
3. A. Tomkins, B. Miklos, G. Corrado, L. Lukacs,´ M. Ganea, P. Young et al., "Smart reply: Automated response suggestion for email," arXiv preprint arXiv:1606.04870, 2016.
4. C. Grevet, D. Choi, D. Kumar, and E. Gilbert, "Overload is overloaded: email in the age of gmail," in Proceedings
5. of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2014, pp. 793–802.
6. E. Sneiders, "Review of the main approaches to automated email answering." in WorldCIST (1), 2016, pp. 135–144.
7. L. Deng, D. Yu et al., "Deep learning: methods and applications," Foundations and Trends R in Signal Processing, vol. 7, no. 3–4, pp. 197–387, 2014.
8. J. M. Ben´ıtez, J. L. Castro, and I. Requena, "Are artificial neural networks black boxes?" IEEE Transactions on neural networks, vol. 8, no. 5, pp. 1156–1164, 1997.
9. G. P. Zhang, "Neural networks for classification: a survey," IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 30, no. 4, pp. 451–462, 2000.
10. R. Bergman, M. Griss, and C. Staelin, "A personal email assistant," Software Technology Laboratory, HP Laboratories, Palo Alto, 2002.
11. V. S. Ayyadurai, "System and method for contentsensitive automatic reply message generation for text-
12. based asynchronous communications," Apr. 6 2004, uS Patent 6,718,368.
13. R. Malik, L. V. Subramaniam, and S. Kaushik, "Automatically selecting answer templates to respond to customer emails." in IJCAI, vol. 7, no. 1659, 2007, p. 3015.

14. R. Mihalcea, C. Corley, C. Strapparava et al., “Corpusbased and knowledge-based measures of text semantic similarity,” in AAAI, vol. 6, 2006, pp. 775–780.
15. L. Kosseim, S. Beauregard, and G. Lapalme, “Using information extraction and natural language generation to answer e-mail,” Data & Knowledge Engineering, vol. 38,
16. no. 1, pp. 85–100, 2001.
17. Y. Watanabe, R. Nishimura, and Y. Okada, “A question answer system based on confirmed knowledge acquired from a mailing list,” Internet Research, vol. 18, no. 2, pp. 165–176, 2008.
18. M. Dredze, T. Brooks, J. Carroll, J. Magarick, J. Blitzer, and F. Pereira, “Intelligent email: Reply and attachment prediction,” in Proceedings of the 13th international conference on Intelligent user interfaces. ACM, 2008, pp. 321–324.
19. J. Martinson, “System and method for eliminating unsolicited junk or spam electronic mail,” Aug. 29 2003, uS Patent App. 10/652,922.
20. A. C. Rothwell, L. D. Jagger, W. R. Dennis, and D. R. Clarke, “Intelligent spam detection system using an updateable neural analysis engine,” Jul. 27 2004, uS Patent 6,769,016.
21. A. Mahendra, “Personified autoresponder.”
22. S. Busemann, S. Schmeier, and R. G. Arens, “Message classification in the call center,” in Proceedings of the sixth conference on Applied natural language processing. Association for Computational Linguistics, 2000, pp. 158–165.
23. A. Bies, M. Ferguson, K. Katz, R. MacIntyre, V. Tredinnick, G. Kim, M. A. Marcinkiewicz, and B. Schasberger, “Bracketing guidelines for treebank ii style penn treebank project,” University of Pennsylvania, vol. 97, p. 100, 1995.
24. “The Stanford Natural Language Processing Group,” https://nlp.stanford.edu/software/lex-parser.shtml/, 2016.
25. “Polar-question dictionary definition — polar-question defined,” http://www.yourdictionary.com/polar-question/.
26. Q. Dictionary, “Questions: wh- questions - English Grammar Today - Cambridge
27. Dictionary,” http://dictionary.cambridge.org/grammar/ british-grammar/questions-and-negative-sentences/ questions-wh-questions/.
28. G. Yule, “Pragmatics/yule g,” Oxford Un. Press, vol. 138,
29. p. 49, 1996.
30. I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” in Advances in neural information processing systems, 2014, pp. 3104– 3112.
31. B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, “Learning deep features for scene recognition using places database,” in Advances in neural information processing systems, 2014, pp. 487–495.
32. S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, “Face recognition: A convolutional neural-network approach,” IEEE transactions on neural networks, vol. 8, no. 1, pp. 98–113, 1997.
33. P. Y. Simard, D. Steinkraus, J. C. Platt et al., “Best practices for convolutional neural networks applied to visual document analysis.” in ICDAR, vol. 3, 2003, pp. 958–962.
34. L. Bottou, “Large-scale machine learning with stochastic gradient descent,” in Proceedings of COMPSTAT’2010. Springer, 2010, pp. 177–186.
35. M. N. Perez, F. J. A. Mata, V. M. Z. Rodriguez, and
36. S. Zhang, “Pervasive healthcare monitoring system,” in Ubiquitous Intelligence and Computing and 2015 IEEE
37. 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UICATC-ScalCom), 2015 IEEE 12th Intl Conf on. IEEE, 2015, pp. 1712–1716.
38. A. Foundation, “Aiml - the artificial intelligence markup language,” http://www.alicebot.org/aiml.html.

39. R. Huddleston, “The contrast between interrogatives and questions,” Journal of Linguistics, vol. 30, no. 2, pp. 411– 439, 1994.
40. L. Bailey, “Question particles: Thai, japanese and english,” Linguistica Atlantica, vol. 32, pp. 34–51, 2013.
41. L. Bailey, “Sentential word order and the syntax of question particles,” Newcastle Working Papers in Linguistics, vol. 16, no. 1, pp. 23–43, 2010.
42. D. Kies, “Modern english grammar,” 2003.
43. M. Selting and E. Couper-Kuhlen, Studies in interactional linguistics. John Benjamins Publishing, 2001, vol. 10.
44. K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “Bleu: a method for automatic evaluation of machine translation,” in Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 2002, pp. 311–318.
45. C. Corley and R. Mihalcea, “Measuring the semantic similarity of texts,” in Proceedings of the ACL workshop on empirical modeling of semantic equivalence and entailment. Association for Computational Linguistics, 2005, pp. 13–18.

Sasa Arsovski is Program Director of AI and Robotic and Associate Professor at Raffles University. Dr. Sasa Arsovski is a highly recognized AI researcher in the field of computer vision in industrial quality control, object detection, image understanding, and genetic algorithms in text processing for neural network chatbots. With more than twenty years of experience as an AI and IT developer, Dr Sasa Arsovski has received a Ph.D. from the Department of Applied Computer Science and Informatics, Faculty of Technical Science, University Novi Sad, Serbia. Since 2004 he has been working as an independent associate for the development of ICT in the Guarantee Fund of AP Vojvodina, as a visiting professor at multiple universities (inc. International University in Novi Pazar, Serbia and University Novi Sad, Serbia). He was also the Lab Vice Director and AI researcher at the Imagineering Institute in Johor.

Adrian David Cheok is an Australian electrical engineer and a professor at iUniversity, Tokyo, Japan. He is among the faculty of Ducere Business School in Prahran, Victoria and is a visiting professor at the University of Novi Sad in Serbia. He is also the director of the Imagineering Institute in Malaysia, the Mixed Reality Lab in Singapore and the CEO of Nikola Tesla Technologies Corporation in Malaysia. Until 2020 Cheok was on the organizing committee of the Love and Sex with Robots conference.

Muniru Idris Oladele is Data Scientist at Classy Advertising, LDA, Idris is a technology enthusiast who strives to learn and understand key functionalities in technological innovations and inventions in order to simplify their applicability for everyday use. Trainer, a R&D enthusiast and programmer with expertise in Artificial Intelligence (AI), Human-Computer Interaction (HCI), data analytics and anticipatory user design (AUD).

Velibor Premcevski (M’94) received the B.S. degree in information technology from University of Novi Sad, Technical faculty “Mihajlo Pupin” Serbia, in 2017 and the M.S. degree in information technology from University of Novi Sad, Technical faculty “Mihajlo Pupin” Serbia, in 2019. He is currently pursuing the Ph.D. degree in computer science at University of Novi Sad, Faculty of Technical Sciences, Novi Sad, Serbia.

From 2018 to 2020, he was a Teaching Associate on Technical faculty “Mihajlo Pupin”, Serbia. His research interest includes the development of application software solutions in various fields as financial, medical, entertainment. From 2020 to now he is Teaching Assistant on Technical faculty “Mihajlo Pupin”, Serbia. Member of the organizing committee of the Applied Internet and Information Technologies International conference AIIT 2019, 2020, 2021.

Branko Markoski is an full professor within the, Techical faculty ‘’M. Pupin’’, in Zrenjanin, University of Novi Sad, Republic of Serbia since 2019. He received his B.Sc. (Dipl. Ing.) degree in 1994, M.Sc. (Magister) degree in 2000 and Ph.D. degree in 2007, all in Computer Science from the University of Novi Sad, Republic of Serbia , Faculty of Technical Sciences. His is author of then more two hundred papers published in journal and different conferences proceedings and participated in more than 20 projects. He has published five books. His research interests include cyber security, Low level programming, system engineering, software architectures and context-aware computing.

Received: November 01, 2021; Accepted: March 01, 2022.

Computer Science and Information Systems 19(2):935–955 https://doi.org/10.2298/CSIS211220019Z

A Consortium Blockchain-Based Information Management System For Unmanned Vehicle Logistics

Manjie Zhai[1], Dezhi Han[1,*], Chin-Chen Chang[2], and Zhijie Sun[1]

[1] College of Information Engineering, Shanghai Maritime University, Shanghai, 201306, China
dzhan@shmtu.edu.cn

[2] The Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan, 000400, China

Abstract. Unmanned vehicle (UDV) delivery technology can meet the special needs of users and realize efficient and flexible distribution of logistics orders. However, there are risks of order data leakage and tampering in the intelligent logistics distribution environment. To solve this problem, this paper designs and implements a system based on the Hyperledger Fabric blockchain platform. Based on the blockchain technology, the system adopts a distributed architecture to establish a secure and trustworthy logistics data management platform to achieve the integrity and traceability of data in the logistics process. The data dual-chain storage strategy is used to ensure the efficiency of data queries. Furthermore, four smart contracts including order management contract (OMC), access control management contract (ACC), access control policy management contract (ACPC), and environmental data management contract (EDC) are designed in combination with the attribute-based access control strategy. By triggering the smart contract, the controllable access of order data can be realized. Finally, two groups of experiments are designed to test the performance of the system. Experimental results show that the proposed system can maintain high throughput in a large-scale request environment under the premise of ensuring data security.

Keywords: Blockchain, Logistics, Attribute-based access control, Hyperledger Fabric, Smart contract.

1. Introduction

With the rapid development of the Internet of things (IoT), the market scale of the logistics industry is gradually expanding. Compared to traditional delivery methods, unmanned vehicle (UDV) delivery can meet the delivery needs of many specific scenarios[26]. For example, during the epidemic period, UDV can realize the non-contact distribution of living materials and medicines, reducing the risk of contact infection. In addition, UDV can also be used for street mail distribution, extreme weather distribution, and other scenarios. However, the existing intelligent logistics platform lacks a complete and reliable credit guarantee system, and problems such as product counterfeiting, loss, and package loss continue to occur[39],[22],[18]. The logistics business is composed of many participants, involving a wide area and a long time span. It is difficult for core institutions to meet the information management and controllable access in the IoT environment. Logistics information is facing the risk of leakage and tampering, and information security is becoming more and more prominent[27],[7],[15],[19].

Blockchain is a distributed data storage and management technology based on a public-key encryption mechanism. The blocks are linked by a hash algorithm as a chain to ensure the integrity and traceability of data, which provides conditions for optimizing logistics management and information traceability[1]. Each distributed node of the blockchain realizes data communication through p2p network and consensus algorithm to ensure data consistency between nodes and secure mutual trust sharing of information[11],[24]. As an open-source consortium chain platform, Hyperledger Fabric inherits the characteristics of the blockchain and also provides a more efficient consensus mechanism, higher throughput, and support for multiple channels[13],[29],[17]. Access control technology is an important means to protect resources and has been widely used in various industries. The traditional access control technology belongs to centralized access control, which has problems such as single-point failure and poor scalability. Traditional access control technology does not meet the access requirements of logistics platform information distribution and mobility[30],[16],[23]. Attribute-based access control (ABAC) is an extension of role-based access control(RBAC). The algorithm extracts the attributes of the user, resource, permission, and environment respectively and flexibly combines these attributes. Finally, the management of permissions is transformed into the management of attributes, which can solve the problem of fine-grained access control that is difficult to be solved by traditional access control and the problem of agent dynamic authorization access in a large-scale environment[8],[31],[20]. ABAC can effectively solve the controllable access of the information in the logistics process, and has a wide range of application scenarios.

To solve the above-mentioned drawbacks, this paper designs a blockchain-based UDV logistics information management system, which mainly focuses on the security and privacy protection of users' data, access control, in the logistics platform. The main contributions of this paper are summarized as follows:

1. We combine Hyperledger Fabric with a distributed architecture to establish a secure and trustworthy logistics data management system to implement the integrity and traceability of data in the logistics.
2. We propose a data storage strategy with dual chains that divides data into order data and real-time environment data, improving the efficiency of data queries.
3. Combined with the ABAC access control algorithm, this paper designs four smart contracts, including the order management contract (OMC), access control management contract (ACC), access control policy management contract (ACPC), and environmental data management contract (EDC). Among them, the OMC manages order information, the ACC manages user access requests, the ACPC manages access policies set by administrators, and the EDC manages environmental data. The ABAC access control policy is deployed on blockchain using a smart contract, and the effective management of logistics data is achieved by triggering a smart contract to ensure controlled access to logistics data.

The rest of this paper is organized as follows. Section 2 introduces the related work. Section 3 introduces the overall design of the system in detail. Section 4 provides the detailed experiment process. Section 5 provides the safety analysis. Finally, Section 6 summarizes the paper.

2. Related Work

2.1. Hyperledger Fabric

Hyperledger Fabric is an open-source distributed ledger platform for enterprise applications [3],[2], using modular structures to provide extensible components. The Fabric includes four types of nodes, namely CA node, Client node, Peer node, and Order node. Client nodes are used to interact with Peers and implement operations such as adding, deleting, and modifying blockchain networks. CA nodes can generate or cancel member identity certificates, providing unified management for the digital certificates of member nodes. Peer nodes are used to store blockchain ledger and chaincode and the application program updates the ledger and checks the chaincode by connecting Peer nodes. The Order node will receive the transactions sent by the Peer node and sort them according to certain rules, and finally package the transactions in a certain order into blocks.

Chaincode: Chaincode is the code deployed on Fabric network nodes to operate and manage the data in the distributed ledger, and is called to implement the smart contract[6].

Channel: Channel in Fabric isolate blockchain data from different organizations, each channel has a proprietary account, and organizational nodes in different channels cannot access directly. Each Peer node in the network needs to be identified by the administrator to join the channel and each communication party must be authenticated and authorized to trade on the channel. This mechanism effectively ensures the security of transaction and improves the utilization of data storage space and parallel processing efficiency[37].

2.2. Attribute-based access control model

Access control technology implements authorized access to resources according to predefined access policies and prevents unauthorized information disclosure by controlling the access rights of the subject to the object[38]. Access strategy is the set of attributes required for specific operations on data resources. The binary group can be expressed as Eq. (1).

$$ABACPolicy \leftarrow \langle AttrSet, Rule \rangle. \tag{1}$$

where *AttrSet* represents the attribute set of the strategy and can be represented as a set of quaternions, as shown in Eq. (2).

$$AttrSet = \langle AS, AO, AE, AP \rangle. \tag{2}$$

AS represents the attribute of the subject, including the identity, role, location, certificate, etc. *AO* represents the resource attributes, including the identity, location, department, type, etc. *AE* represents the attribute of the environment, which is used to judge whether the policy is satisfied the request. *AP* represents the attribute of permission, which means the operation of the subject on the object, such as write, modify, delete, etc. The *Rule* represents a set of rules that can be expressed as $Rule = \{rule_1, rule_2, \ldots, rule_n\}$, $n \geq 1$, $rule_n$ denotes the nth rule. *Rule* can be expressed as a set of quaternions as Eq. (3) and Eq. (4).

$$Rule = Result = F(.). \tag{3}$$

$$F(Attr(S_i), Attr(O_i), Attr(E_i), Attr(P_i)) \rightarrow \{Permit, Deny\}. \tag{4}$$

Eq. (4) indicates that the subject with the authorization attribute S_i performs an access action with the attribute value P_i to the object O_i in the context of the environment attribute E_i.

2.3. Elliptic curve digital signature method

The elliptic curve digital signature algorithm is based on the elliptic curve algorithm and signature algorithm, which is mainly used to create digital signatures for data. It has the characteristics of identifiability and unforgeability, ensuring that the authenticity of the data is verified without destroying the security of the data [34],[28]. The elliptic curve digital signature method is constructed based on Eq. (5).

$$y^2 = (x^3 + a \times x + b) \; mod \; p. \tag{5}$$

The parameters of the elliptic curve are (a, b, p, n, G), where a and b are the parameters of the curve equation, p is the base of the modular operation, n is the number of points on the curve, the parameter G represents the selected reference starting point that can be any point on the curve, and the key pair is (SK, PK), where SK is the private key and PK is the public key.

Signature phase:

1. Generate a random number k, which satisfies the condition: $1 \leq k \leq n-1$.
2. Compute $p = k \times G$, the abscissa of p is R.
3. Compute $r = R$ **mod** n. if $r = 0$, return to Step 1.
4. Calculate the hash $H(m)$ of message m and convert the obtained value into a large integer z.
5. Calculate s using $s = k^{-1}(z + SK \times r)$ **mod** p . if $s = 0$, return to Step 1.
6. (r, s) is the signature of the message m.

Verification phase:

1. Calculate $H(m)$ and convert it to integer Z.
2. Calculate w, where $w = s^{-1}$ **mod** n.
3. Calculate $u_1 = (z \times w)$ **mod** n, $u_2 = (r \times w) \; mod \; n$.
4. Calculate $X = (x_1, \; y_1) = u_1 G + u_2 PK$. If $X = 0$, the verification is wrong; otherwise, convert the abscissa of X to R and calculate $v = R$ **mod** n.
5. If $v = r$, the verification passes.

3. System overall design

This paper proposed a UDV logistics information management system based on the alliance chain, solving the security and privacy protection issues of data in the logistics platform and ensuring the controllable access of logistics data. This section introduces the overall design of the system. Tab. 1 lists the key symbols used in this paper. Fig. 1 is a framework diagram of the system model. The infrastructure layer is the Fabric underlying module. The data storage layer includes order data storage module and real-time environmental data storage module. The smart contract layer includes four smart contracts and the corresponding state database key-value pair storage. The user interaction layer includes the UDV registration and management module, order allocation module, real-time navigation, and ABAC module.

Table 1. Key symbol description.

Notations	Description
$Cert_i$	Certificate of entity
PK_i	Public key of $user_i$
SK_i	Private key of $user_i$
$Info_i$	Identity information of $entity_i$
$TimeStamp_i$	Timestamp of $entity_i$
Sign(.)	Signature algorithm
ABACPolicy	Attribute-based access control policy
OMC	Order management contract
ACC	Access control management contract
ACPC	Access control policy management contract
EDC	Environmental data management contract
UDV	Unmanned vehicle
CA	Authority
DockerImage	Docker image
Config(.)	Config file of the node
Start(.)	Start of the node
Install(.)	Install of the chaincode

3.1. Entity description

1. **UDV**: It is the transportation mode for delivering goods. UDV should register its identity information with CA and obtain order distribution qualification before distribution.
2. **User**: There are two types of users including data requester and order owner.
3. **CA**: As the system administrator, CA is responsible for system initialization and entity authentication[4],[35]. Any entity that wants to join the blockchain should register its identity information and obtain PK and SK, and only entities certified by CA can perform operations such as uploading and querying data.
4. LC: The LC reasonably allocates logistics orders to UDV according to *OrderDistribute()* in OMC.
5. Sensor: The Sensor is an important configuration on the UDV, which can sense vehicle position change, navigate the route, improve communication reliability, and ensure low latency interaction of measurement information.

3.2. Data storage

This section introduces the double-chain storage strategy in detail. Logistics data is divided into order data and real-time environment data. Order data refers to the information of receiving and delivering users and commodity-related information in the logistics, and the fields included are shown in Tab. 2. On the blockchain, the order number of the order data is taken as the key value, the order creation time, the sender's and receiver's information, the order information, and other fields as the value of the map after JSON serialization.

Environmental data refers to real-time data such as road conditions collected by sensor equipment. When the UDV encounters a failure, the surrounding road condition status and the information of the nearby UDV can be obtained by querying the environmental

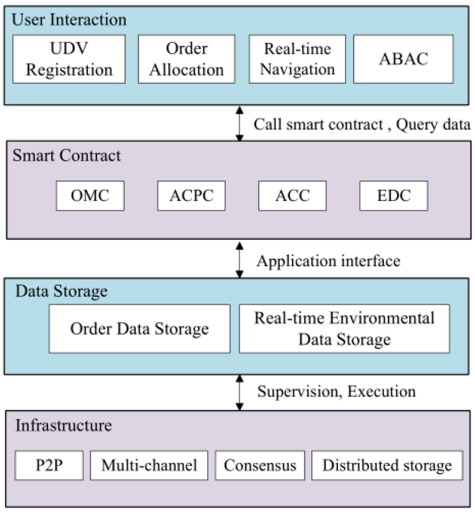


Fig. 1. System architecture diagram.

Table 2. Order data field description.

Notations	Description
OrderId	Order number
OrderTime	Order create time
SenderAddress	Sender address
SenderPhone	Sender telephone
ReceiverName	Receiver name
ReceiverAddress	Receiver Address
ReceiverPhone	Receiver telephone
ReceiverReput	Receiver reputation
OrderState	Order state
Private	Order private
Urgent	Order urgency

data information to prepare for the order handover[12]. The data contained in the environmental information is shown in Tab. 3. Environmental data includes the IOT module and the message queue module. The IoT module mainly through sensor devices to collect environmental data such as vehicle speed, road condition status, and nearby UDV information. To reduce the error of the data, the original data collected 30 times per second are taken as a group, and its average value is stored in the environmental data link. The smart contract saves the data by calling the relevant interface. If the data can be successfully written into the ledger, the corresponding message will be dequeued. otherwise, the data will be temporarily stored in the queue for the next data uplink. The environmental data uplink process is shown in Fig. 2.

Table 3. Environment data field description.

Notations	Description
IotId	Internet of things device number
VehDataTime	Vehicle data acquisition time
VehDataAddress	Vehicle data acquisition address
VehSpeed	Vehicle speed
RoadState	Road condition status
NeighUdvNum	Number of nearby UDV
NeighUdvState	Status of nearby UDV
NeighUdvSpeed	Speed of nearby UDV

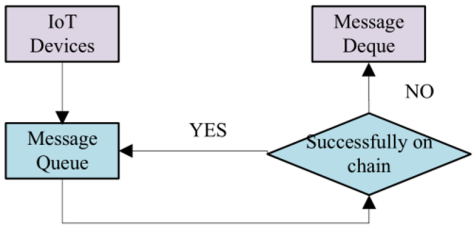


Fig. 2. Environment data uplink process.

3.3. Smart Contract

OMC. OMC is to manage the order data according to the user's request, including the request situation of order addition, order cancellation, order information change, and order handover. The methods included in the OMC are as follows.

AddOrder(): When a user request to add an order, *CheckOrder()* in the OMC is triggered to check the rationality of the order request. If the order request is reasonable, the order will be released and the order information will be added to the status database, and the order operation record will be written to the blockchain. The pseudocode of *AddOrder()* is shown in Algorithm 1, where the *OrderId* is stored as the key, and *OrderTime, SenderAddress, SenderPhone* and other attributes are stored as value in the blockchain.

QueryOrder(): It realizes the function of querying the details of order data according to *OrderId*.

UpdateOrder(): In some special cases, the information needs to be modified after the user places an order, which can be processed according to the order status. If the order is not issued, *UpdateOrder()* can be called to complete the information modification operation; otherwise, it cannot be modified.

DeleteOrder(): when the user sends a request to delete an order, *CheckOrder()* is first triggered to check the rationality of the order information, and then *QueryOrder()* is called to check the order status. If the order is not issued, *DeleteOrder()* is called to delete the relevant information in the order data chain. Otherwise, the wireless sensor device should send the withdrawal or cancellation command to the UDV in time, and change the order status information to undelivered.

CheckOrder(): It is used to check the reasonableness of key field information when the users submit order requests.

Algorithm 1: OMC.*AddOrder()*

Input: *Request*(*AddOrder*)
Output: *Success* or *Error*

$APIstubChaincodeStub \leftarrow Invoke()$;
if $CheckOrder(.) == False$ **then**
  return $Error(Illeagleorder)$;
end
$Id \leftarrow sha256(UserId, OrderId)$;
$ans \leftarrow APIstub.PutState(Id)$;
if $ans \neq null$ **then**
  return $Success$;
end
return $Error$;

Algorithm 2: OMC.*DeleteOrder()*

Input: *Request*(*DeleteOrder*)
Output: *DeleteResult* or *Error*

$APIstubChaincodeStub \leftarrow Invoke()$;
if $CheckOrder(.) == False$ **then**
  return $Error$;
end
$ans == QueryOrder(OrderId)$;
if $ans == NoIssue$ **then**
  $DeleteOrder(OrderId)$;
end
$Revoke(OrderId)$;

Checkudvinfo(): It is used to verify the identity and status information of UDV applying for distribution qualification.

OrderDistribute(): After the UDV is qualified for order distribution, the LC will distribute the order reasonably, then update the order status in time and store it in the order chain.

OrderCharge(): Before the order is delivered, the shipper and the user will negotiate the charging criteria. If the goods are lost, damaged, or the order is delivered over time during the delivery process, the compensation should be paid by the agreed compensation criteria. After the user successfully obtains the goods, *OrderCharge()* will be called to complete the payment operation.

ACPC. The ACPC is to provide management functions for the established attribute-based access control strategy. Combining the characteristics of logistics data and the attribute-based access control model, this paper defines the attribute characteristics as follows:

P = {*AS, AO, AP, AE*},
AS = {*UserId, Role, PK, UserGroup*},
AO = {*OrderId, Singer, SignOrderData*},
AP = {*OrderPermission*},
AE = {*CreateTime, EndTime, Address, CurrentAccess*}.

AS mainly refers to the attribute of users who access data resources, including *UserId, Role, PK*, and *UserGroup*, where *UserId* is the unique identification of user information.

AO mainly refers to the order data stored in the order chain, including *OrderId, Singer*, and the signature of order data *SignOrderData*.

AP includes the order data access permission attribute *OrderPermission*.

AE refers to the environmental attribute of the order requester, including *CreateTime, EndTime, Address*, and *CurrentAccess*.

Administrator formulates access policies based on user, resource, operation, and environment attributes, as shown in Eq. (6).

$$f(AS,\ AO,\ AP,\ AE) \rightarrow ABACPolicy \tag{6}$$

Add the policy to the SDB by calling *AddPolicy()* in the ACPC. *VerifyPolicy()* is used to verify whether the access policy formulated by the administrator is a legal policy, *QueryPolicy()* supports querying policies through *AS* or *AO* features, and *UpdatePolicy()* is invoked to update the operation records on the blockchain. When the specified access policy exceeds the specified period, *DeletePolicy()* is called to delete.

Algorithm 3: ACC.*CheckAccess()*

Input: *ABACRequest*
Output: *Success* or *Error*

$\langle A_uS,\ A_uO,\ A_uE\rangle \leftarrow GetAttrs(ABACRequest)$;
$P = \langle P_1,\ P_2,\ ...,\ P_n\rangle \leftarrow ACPC.QueryPolicy(A_uS,\ A_uO)$;
if $P == Null$ **then**
  return *Error("This is an illeagle policy")*;
end
for P *in* $\langle P_1,\ P_2,\ ...,\ P_n\rangle$ **do**
  $\langle ...,\ A_PP,\ A_PE\rangle \leftarrow P$;
  if $Value(A_PP) == deny$ **then**
    $continue$;
  end
  $ans \leftarrow QueryOrder()$;
end
if $ans \neq Null$ **then**
  return *Success("OK")*;
end
return *Error("UnAccess")*;

ACC. ACC is used to verify whether the user's data access request meets the access control policy formulated by the administrator, including the following methods.

Auth(): The main function is to use the user public key to verify the authenticity of user identity. When the user sends data access requests, the user will use the ACC's public key to encrypt the data in the request, and then sign the request with his own private key. After receiving the request, ACC calls the *Auth()* method and uses public key to verify user's identity, and then uses its own private key to decrypt. The successful decryption means that the user's access request is reasonable and the data can be successfully accessed.

GetAttrs(): After verifying the user's identity, parse the attribute fields contained in the user request by calling *GetAttrs()*. The request contains subject attributes and objects attributes $\{AS, AO\}$.

CheckAccess(): First, get the attribute $\{AS, AO\}$ through *GetAttrs()*, and then call *QueryPolicy()* of the ACPC to query the corresponding attribute access control policy according to *AS* and *AO*. If the query result is null, there is no policy to support the request. Otherwise, one or more access policies will be obtained, and then judge whether the attribute value *AE* in the request matches the *AE* in the access policy and whether the value of *OrderPermission* in the *AP* is 1. If all attributes match the policy, the verification passes. Finally, the related functions in the OMC are invoked to complete the access to the order resources, update, and delete operations. Otherwise, return the request fails. Algorithm 3 is the pseudocode of *CheckAccess()*.

EDC. The EDC includes the environmental data uplink *AddEnvData()* and *GetEnvData()* to obtain environmental data.

AddEnvData() is similar to *AddOrder()* in Algorithm 1. The *AddEnvData()* method implements the interface for inserting data, adds environmental data to the state database, and updates the operation record to the environmental data chain.

The main function of *GetEnvData()* is to obtain environment data and query the corresponding environment data from the database based on the IoT ID.

3.4. Workflow

This section details the system workflow, as shown in Fig. 3, including five stages, entity registration, the user placing an order, granting UDV order delivery qualifications, order allocation, delivery, and user request.

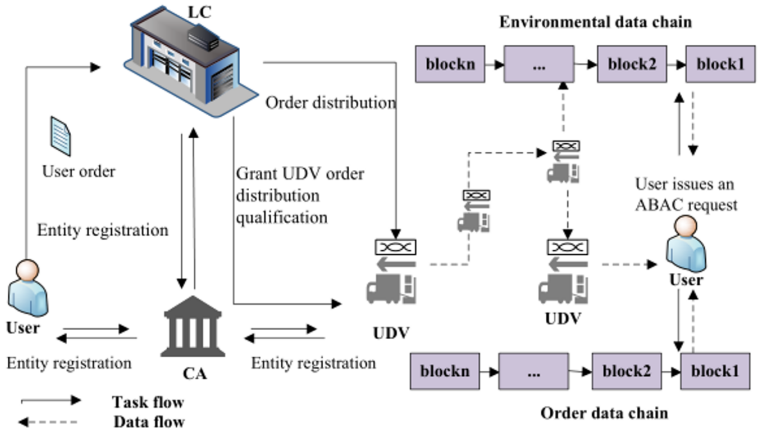


Fig. 3. Environment data uplink process.

Entity registration. The entity requiring authentication sends identity information to CA through the secure channel. CA encrypts the received entity identity information with PK

and then issues a certificate to the entity. As shown in Eq. (7), $Info_i$ is the entity identity, $Sign_{CA}$ is the CA's signature for the entity, and $TimeStamp_i$ is the corresponding timestamp.

$$CA \rightarrow Certi\{PK_i,\ Info_i,\ Sign_{CA},\ TimeStamp_i\} \tag{7}$$

User Order. Users order according to their own needs, and the system will create a transaction order according to the order time, recipient address, and contact information.

$$OMC(AddOrder()) \rightarrow \{Ledger,\ SDB\} \tag{8}$$

Grant UDV order distribution qualification. After registration, UDV needs to obtain the order delivery qualification granted by CA. Firstly, verify the identity of the UDV through *Auth()* in the ACC, and then the UDV sends a distribution request to CA. Once receiving the request, CA checks *UdvCert, UdvState, Distance* by calling *CheckUdvInfo()* in the OMC, the UDV distribution qualification can be granted after passing the verification. The process is shown in Algorithm 4.

Algorithm 4: Grant UDV order delivery qualification

Input: *Request(Delivery)*
Output: *Success* or *Error*

```
if Auth(Udv) == False then
    return Error("UnAuthorized");
end
ans ← CheckUdvInfo(UdvId, UdvCert, UdvState, Distance);
if ans ≠ Null then
    return Error("UnAuthDelivery");
end
return Success("OK");
```

Order Distribution. LC calls *OrderDistribute()* in the OMC, and refer to *OrderTime*, *UdvCert*, and *UdvState* to allocate orders to UDV and prepare for the delivery task reasonably. After the UDV finishes loading, it starts to distribute the goods.

$$OrderDistribute(OrderId,\ OrderTime,\ UdvCert,\ UdvState) \rightarrow Udv \tag{9}$$

During the delivery process, the sensor hardware device configured on the UDV will collect environmental data in real-time, and upload the data to the environmental data chain by calling *AddEnvData()* in the EDC.

$$EDC(AddEnvData()) \rightarrow \{Ledger,\ SDB\} \tag{10}$$

User request. Users can query order information in real-time according to their own needs, which ensures that they have a dynamic understanding of the entire order distribution process. When the goods arrive at the agreed place, the sensor will send the current location information to the blockchain node nearest to the UDV and notify the user that

the order has been delivered[21]. When the pick-up user arrives at the destination, the order can be successfully obtained after being verified as a legitimate user.

$$ACC(Request\{AS,\ AO,\ AP\}) \rightarrow \begin{cases} 1, \textit{Permit} \\ 0, \textit{Deny} \end{cases} \tag{11}$$

The user initiates a request to access the order data. Once receiving the request, the blockchain triggers the smart contract and calls *Auth()* in the ACC to verify the authenticity of the user's identity.

$$Request\{AS,\ AO,\ AP\} \rightarrow BlockChain \tag{12}$$

$$ACPC(Auth(AS,\ AO,\ AP,\ PK_i)) \rightarrow Entrpted\_ABACPolicy \tag{13}$$

If the verification passes, *QueryPolicy()* in the ACPC will be invoked to query the ACC based on the subject and object attributes. If the result is not empty, one or more access policies are obtained. Otherwise, there is no policy to support the request.

$$Decrypte(Entrpted\_ABACPolicy, SK_i) \rightarrow \langle ABACPolicy, OK \rangle \tag{14}$$

$$ACPC(QueryPolicy(AS,\ AO)) \rightarrow \begin{cases} one\ or\ more\ policy, & OK = 1 \\ null, & OK = 0 \end{cases} \tag{15}$$

If all attributes match the policy, the relevant methods in the OMC or EDC are invoked to complete the operation of accessing, updating and deleting logistics data.

$$OMC \xrightarrow{add/delete/query/update} OrderData \tag{16}$$

$$EDC \xrightarrow{add/query} EnvData \tag{17}$$

4. Experiment

The experiment was completed on a single computer. The CPU model of the computer is i7-6700 and the memory size is 8G. The software environment required for the experiment is shown in Tab. 4. The experimental steps include configuration fabric network, system initialization and startup steps, chaincode installation, system function test, and system throughput test. The functional test mainly tests a series of operations on the order data chain, such as user querying order information, modifying order information, etc. For the environment data, *AddEnvData()* and *QueryEnvData()* is only tested because the amount of the environment data is large and the query is only used when the order is handed over in the case of the failure of the UDV.

Two groups of experiments were conducted. The first group tested the throughput of four smart contracts when the numbers of concurrent requests were 10, 50, 100, 200, 300, 400, 500, 600, 700, 800, 900, and 1000. The second group of experiments tested the time spent on four smart contracts with 10-1000 concurrent requests, respectively.

Table 4. Environment data field description.

Software and System	Version
OS	Ubuntu 16.04
Hyperledger Fabric	v1.4.3
Docker	v18.09.7
Docker-compose	v1.8.0
Node	v12.18.3
Golang	v1.14.6
Git	v2.7.4

Table 5. Environment data field description.

Environmental Parameters	Value
Couchdb	4
CA	2
Orderer	1
Peer	4
Fabric-tools	1
Fabric-iot/chaincode	16

4.1. System Construction

Tab. 5 is the composition of experimental environment nodes. The experimental steps include system environment initialization, system startup, chaincode install, and update.

System initialize: Firstly, the binary tools provided by the Hyperledger Fabric are used to generate certificates and key pairs for Order and Peer nodes of different organizations. Secondly, move the node's certificate and key pair to the file directory mounted on the docker image of the CA node. When the CA container is started, other nodes can authenticate their identities with the signature.

$$CA \rightarrow \{Certpeer,\ Certpeer,\ Certchannel,\ PK_i,\ SK_i\} \tag{18}$$

$$Build(Certi,\ PK_i,\ SKi) \rightarrow DockerImage \tag{19}$$

System startup: Use the configtxgen tool provided by Hyperledger Fabric to generate the genesis block [25] and configure the channel. Write the configuration information of nodes and channels into transactions to ensure that the identity information of each section in the system can be traced and tampered with. Blockchain starts nodes based on the docker container and Hyperledger. Each node has an independent environment, communicates with each other through port calls, and executes the blockchain to the startup script, to quickly realize a series of processes such as creating channels and joining channels.

$$Config(Cert_i,\ PK_i,\ SK_i) \overset{configtxgen}{\longrightarrow} Transaction \tag{20}$$

$$Start(Docker,\ Fabric) \overset{join}{\longrightarrow} Channel \tag{21}$$

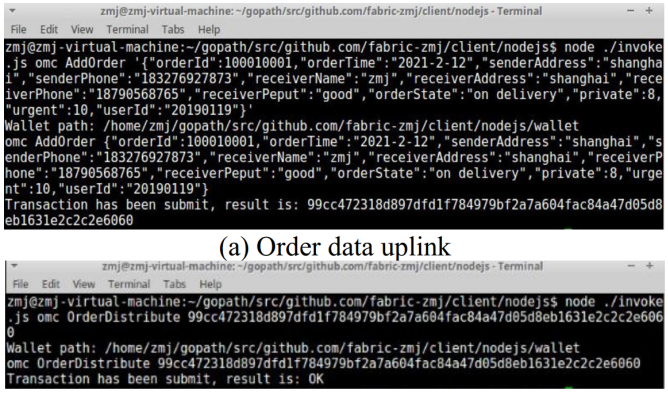
Chaincode install and update: After the system starts successfully, chaincode starts to install. Since the client's image has attached the directory of chaincode, chaincode can be compiled directly in the client's image. The quick installation of chaincode is

realized by executing the script install.sh, and the quick update of chaincode is realized by executing the script upgrade.sh. Once the chaincode is successfully installed, the order storage, user access and strategy formulation process are tested.

$$Install(Chaincode) \overset{Client}{\longrightarrow} Transaction \tag{22}$$

4.2. Function testing

Order stored procedure. First, add the order data to the blockchain by calling *AddOrder()*, and then call *OrderDistribute()* to allocate the order stored in the blockchain reasonably. Fig. 4 is the order stored procedure.



(a) Order data uplink

(b) Order allocation

Fig. 4. Order storage process.

Fig. 5. Order storage process.

User access process. Fig. 5 shows calling CheckAccess() to access order data.

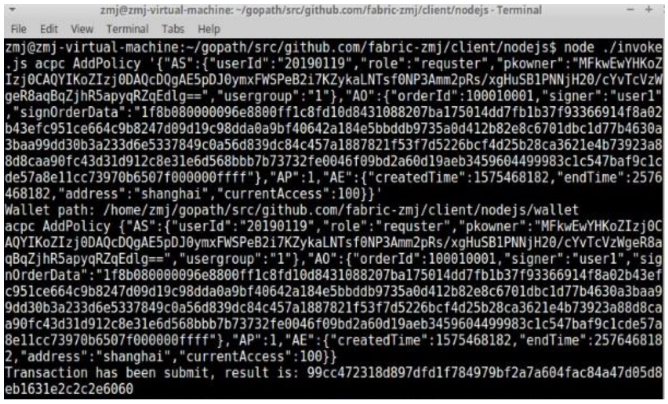
Access control policy development process. The access control policy is shown in Fig. 6. The administrator formulates relevant access policies based on *UserId* and *OrderId* and adds the access policies to the blockchain network through *AddPolicy()*.

4.3. Throughput testing

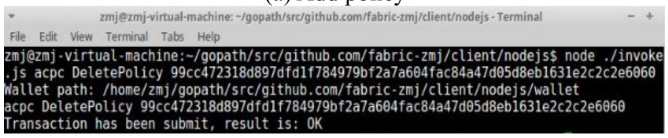
Throughput is an important index to evaluate system performance that is affected by software and hardware device, block size, and consensus algorithm. In the blockchain, throughput refers to the number of transactions that can be processed per unit time.

$$TPS = \frac{Transactions\ concurrency}{Average\ response\ time} \tag{23}$$

To test the system performance, the throughput of four smart contracts with 10, 50, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000 concurrent requests are tested in the first group of experiments. To avoid the impact of uncertain factors such as network downtime, under different concurrent requests, ten groups of throughput data with relatively stable network status are selected respectively and the average value is taken as the final throughput. The test results are shown in Fig. 7.



(a) Add policy



(b) Delete policy

Fig. 6. Access control policy development process.

In the second group of experiments, the same method was used to test the average time spent by the four smart contracts in the case of 10-1000 requests. The test results are shown in Fig. 8.

Fig. 7 shows that when the OMC is executed, the throughput of *DeleteOrder()* is the highest and that of *UpdateOrder()* is the lowest as the number of transactions increases. The throughput of *OrderDistribute()* finally stabilized at 175-180. Since both *OrderDistribute()* and *AddOrder()* contain a read operation and a write operation, the throughput

is almost equal. While *QueryOrder()* contains only one read operation, its throughput is higher than *OrderDistribute()* and *AddOrder()* is stable in the range of 200-205.

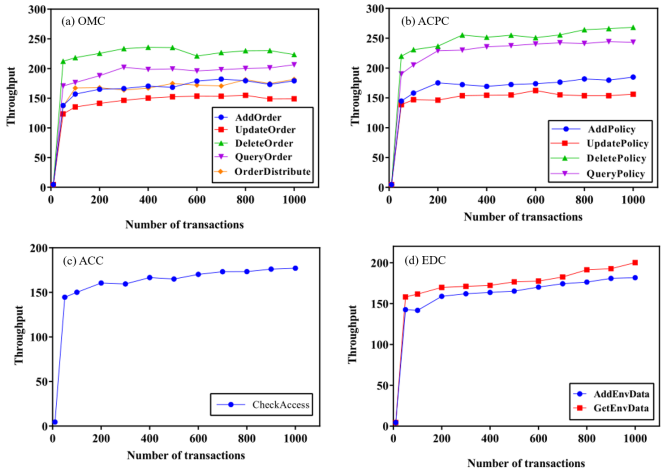


Fig. 7. Access control policy development process.

When the ACPC is executed, the throughput of *DeletePolicy()* is the highest and *UpdatePolicy()* is the lowest as the number of transactions increases. Since *DeletePolicy()* does not generate transactions and can directly delete existing transactions in the database, the throughput is high and finally stabilizes at 260-265 as the number of transactions increases. *QueryPolicy()* contains only one read operation, its throughput is high and finally stabilizes at 240-245. *AddPolicy()* contains one read operation and one write operation and its throughput is finally stable at 180-185. Compared with *AddPolicy()*, *UpdatePolicy()* contains one read operation and two write operations, its throughput is lower than 180 and stable at 150-155.

When the ACC is executed, the throughput of *CheckAccess()* is finally stabilized at 175-180. Since *CheckAccess()* calls *QueryPolicy()* in the ACPC and *CheckAccess()* includes a read operation and a write operation, so its throughput is lower than *QueryPolicy()* in the ACPC.

When the EDC is executed, the throughput of *AddEnvData()* is finally stabilized at 170-175. *GetEnvData()* calls *QueryPolicy()* in the ACPC, and it only includes one read operation with its throughput finally stabilized at 185-190, which is higher than *AddEnvData()* but lower than *QueryPolicy()*.

Fig. 8 shows the average time spent by ACPC, OMC and EDC in the case of 10-1000 requests. As shown in Fig. 8(a), as the number of transactions increases, the time spent continues to increase. In general, the time spent by each method is as follows:

$DeletePolicy() < QueryPolicy() < AddPolicy() < UpdatePolicy()$

The time spent by each method in Fig. 8(b) is as follows:

$DeleteOrder() < QueryOrder() < AddOrder() \approx OrderDistribute() < UpdateOrder()$

In Fig. 8(c), *AddEnvData()* takes more time than the *GetEnvData()*. After analysis, software and hardware devices, the number of nodes, block size, and consensus algorithm will affect the reading and writing speed of the blockchain. Writing operation usually

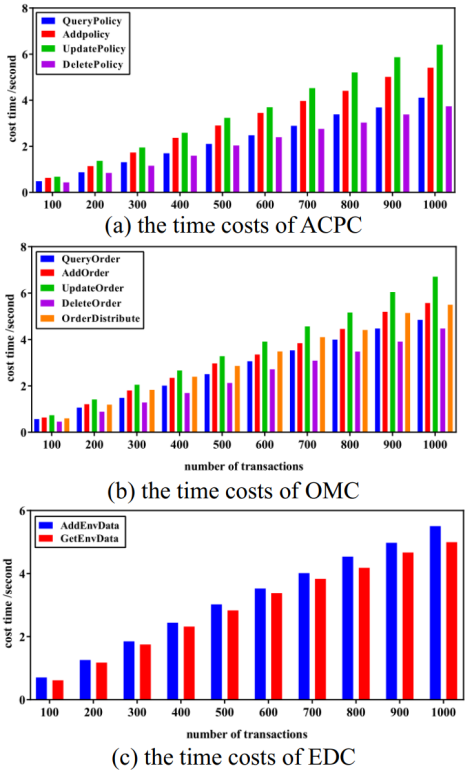


Fig. 8. Access control policy development process.

involves creating new blocks, generating new transactions, etc., which takes more time. Experiments show that the read operation takes less time than the write operation. The system throughput increases as the number of requests increases, and the throughput tends to be stable when the throughput reaches a certain value. With the further increase in the number of customer requests, there is no significant decline in throughput.

The experimental results are in line with expectations. The system can maintain high throughput in a large-scale request environment, realize dynamic fine-grained access control of logistics information, and meet the operation requirements of the actual logistics platform for data information.

5. Safety analysis

Entity authentication: Assuming that a malicious entity attempts to impersonate a legitimate user since each legitimate user obtains the unique certificate issued by CA, the system can verify the user's identity according to the user certificate [32],[33]. If the user certificate is not disclosed, the malicious entity cannot obtain any valid data information through counterfeiting. Therefore, entity authentication can effectively confirm the identity of the entity and ensure that the order data cannot be leaked.

Data integrity: All order records in the system are stored after being signed by using the principle of asymmetric cryptography. The data requester verifies the correctness of the data source after obtaining the order data. Since all records are stored on the blockchain, the decentralized environment provided by the blockchain can ensure that they will not be tampered with by any malicious entity[10],[5]. Therefore, data integrity can be guaranteed.

Data access security: When the data requester sends a data access request, the data can be accessed only when its attributes meet the access policy[36], [9]. Therefore, it can ensure that only authorized users can obtain order data, realizing controllable access to order data and the security of order data.

Information traceability: Suppose the blockchain is $BC = bc_1, bc_2, bc3, \ldots, bc_n$, where $bc_i(1 < i < n)$ is the ith block, tx_{ij} represents the jth record in the ith block, and tx_{ijk} represents the information of order k corresponding to the jth transaction in the ith block. According to the order information k, the order record can be obtained through a query on the order data chain to further obtain specific information such as order number[14]. Since the block header contains timestamp proof, the corresponding order record information can be queried through the order number, and the earliest order creation record can be traced according to the timestamp order. To sum up, the order information in this system can be tracked and queried.

6. Conclusions and future work

Based on the Hyperledger Fabric platform, this paper designs and implements the UDV logistics information management system based on the alliance chain, which effectively solves the problems of lack of trust, data leakage, and controllable access. Using the advantages of blockchain technology, such as decentralization, traceability, and non-tampering, ABAC is deployed on the blockchain by designing smart contracts to ensure data integrity, traceability, and controllability. At the same time, the dual-chain storage strategy is designed to alleviate the pressure of the main chain and ensure the efficiency of data queries. Finally, the experimental results prove that the scheme can meet the operation requirements of data information in the actual logistics platform, realize the dynamic fine-grained access control of logistics information, and ensure the security of data. In summary, the system is effective and feasible for the storage of logistics data and access control management. Future work will try to improve the following two aspects:

1. Consider using more physical equipment to test the reliability and throughput of the system.
2. Consider the reputation of UDV and the LC in the process of logistics distribution.

Acknowledgments. This work was supported by the National Natural Science Foundation of China under Grant 61672338 and Grant 61873160.

References

1. Ahmad, R.W., Hasan, H., Jayaraman, R., Salah, K., Omar, M.: Blockchain applications and architectures for port operations and logistics management. Research in Transportation Business & Management p. 100620 (2021)

2. Baliga, A., Solanki, N., Verekar, S., Pednekar, A., Kamat, P., Chatterjee, S.: Performance characterization of hyperledger fabric. In: 2018 Crypto Valley conference on blockchain technology (CVCBT). pp. 65–74. IEEE (2018)
3. Benhamouda, F., Halevi, S., Halevi, T.: Supporting private data on hyperledger fabric with secure multiparty computation. IBM Journal of Research and Development 63(2/3), 1–8 (2019)
4. Cui, M., Han, D., Wang, J.: An efficient and safe road condition monitoring authentication scheme based on fog computing. IEEE Internet of Things Journal 6(5), 9076–9084 (2019)
5. Cui, M., Han, D., Wang, J., Li, K.C., Chang, C.C.: Arfv: an efficient shared data auditing scheme supporting revocation for fog-assisted vehicular ad-hoc networks. IEEE Transactions on Vehicular Technology 69(12), 15815–15827 (2020)
6. Dai, W., Wang, Q., Wang, Z., Lin, X., Zou, D., Jin, H.: Trustzone-based secure lightweight wallet for hyperledger fabric. Journal of Parallel and Distributed Computing 149, 66–75 (2021)
7. Gu, Q., Fan, T., Pan, F., Zhang, C.: A vehicle-uav operation scheme for instant delivery. Computers and Industrial Engineering 149, 106809 (2020)
8. Han, D., Pan, N., Li, K.C.: A traceable and revocable ciphertext-policy attribute-based encryption scheme based on privacy protection. IEEE Transactions on Dependable and Secure Computing PP(99), 1–14 (2020)
9. Han, D., Zhu, Y., Li, D., Liang, W., Souri, A., Li, K.C.: A blockchain-based auditable access control system for private data in service-centric iot environments. IEEE Transactions on Industrial Informatics 18(5), 3530–3540 (2022)
10. Hang, L., Kim, D.H.: Design and implementation of an integrated iot blockchain platform for sensing data integrity. Sensors 19(10), 2228 (2019)
11. Huang, J., Kong, L., Chen, G., Wu, M.Y., Liu, X., Zeng, P.: Towards secure industrial iot: Blockchain system with credit-based consensus mechanism. IEEE Transactions on Industrial Informatics 15(6), 3680–3689 (2019)
12. Huang, K., Wen, M., Park, J., Sung, Y., Cho, K.: Enhanced image preprocessing method for an autonomous vehicle agent system. Computer Science and Information Systems 18(2), 461–479 (2021)
13. Islam, M.A., Madria, S.: A permissioned blockchain based access control system for iot. In: 2019 IEEE International Conference on Blockchain (Blockchain). pp. 469–476. IEEE (2019)
14. Kamble, S.S., Gunasekaran, A., Sharma, R.: Modeling the blockchain enabled traceability in agriculture supply chain. International Journal of Information Management 52, 101967–101978 (2020)
15. Kuru, K., Ansell, D., Khan, W., Yetgin, H.: Analysis and optimization of unmanned aerial vehicle swarms in logistics: An intelligent delivery platform. Ieee Access 7, 15804–15831 (2019)
16. Li, D., Han, D., Crespi, N., Minerva, R., Sun, Z.: Fabric-scf: A blockchain-based secure storage and access control scheme for supply chain finance (2021)
17. Li, D., Han, D., Liu, H.: Fabric-chain chain: A blockchain-based electronic document system for supply chain finance. In: Zheng, Z., Dai, H.N., Fu, X., Chen, B. (eds.) Blockchain and Trustworthy Systems. pp. 601–608. Springer Singapore, Singapore (2020)
18. Li, D., Han, D., Zheng, Z., Weng, T.H., Li, H., Liu, H., Arcangelo: Moocschain: A blockchain-based secure storage and sharing scheme for moocs learning. Computer Standards Interfaces 81, 103597 (2022)
19. Li, H., Han, D., Tang, M.: A privacy-preserving storage scheme for logistics data with assistance of blockchain. IEEE Internet of Things Journal (2021)
20. Li, J., Chen, X., Chow, S.S., Huang, Q., Wong, D.S., Liu, Z.: Multi-authority fine-grained access control with accountability and its application in cloud. Journal of Network and Computer Applications 112, 89–96 (2018)
21. Liang, W., Xie, S., Cai, J., Xu, J., Hu, Y., Xu, Y., Qiu, M.: Deep neural network security collaborative filtering scheme for service recommendation in intelligent cyber-physical systems. IEEE Internet of Things Journal pp. 1–1 (2021)

22. Liu, H., Han, D., Li, D.: Behavior analysis and blockchain based trust management in vanets. Journal of Parallel and Distributed Computing (2) (2021)
23. Liu, H., Han, D., Li, D.: Fabric-iot: A blockchain-based access control system in iot. IEEE Access 8, 18207–18218 (2020)
24. Mohanty, S.N., Ramya, K., Rani, S.S., Gupta, D., Shankar, K., Lakshmanaprabu, S., Khanna, A.: An efficient lightweight integrated blockchain (elib) model for iot security and privacy. Future Generation Computer Systems 102, 1027–1037 (2020)
25. Nakamoto, S.: Bitcoin: A peer-to-peer electronic cash system. Decentralized Business Review pp. 260–268 (2008)
26. Neghabadi, P.D., Samuel, K.E., Espinouse, M.L.: Systematic literature review on city logistics: overview, classification and analysis. International Journal of Production Research 57(3-4), 865–887 (2018)
27. Ni, H., Deng, X., Gong, B., Wang, P.: Design of regional logistics system based on unmanned aerial vehicle. In: 2018 IEEE 7th Data Driven Control and Learning Systems Conference (DDCLS). pp. 1045–1051. IEEE (2018)
28. Nyame, G., Qin, Z., Obour Agyekum, K.O.B., Sifah, E.B.: An ecdsa approach to access control in knowledge management systems using blockchain. Information 11(2), 111 (2020)
29. Outchakoucht, A., Hamza, E., Leroy, J.P.: Dynamic access control policy based on blockchain and machine learning for the internet of things. Int. J. Adv. Comput. Sci. Appl 8(7), 417–424 (2017)
30. Qiu, J., Tian, Z., Du, C., Zuo, Q., Su, S., Fang, B.: A survey on access control in the age of internet of things. IEEE Internet of Things Journal 7(6), 4682–4696 (2020)
31. Rahman, M.U., Guidi, B., Baiardi, F.: Blockchain-based access control management for decentralized online social networks. Journal of Parallel and Distributed Computing 144, 41–54 (2020)
32. Shen, M., Liu, H., Zhu, L., Xu, K., Yu, H., Du, X., Guizani, M.: Blockchain-assisted secure device authentication for cross-domain industrial iot. IEEE Journal on Selected Areas in Communications 38(5), 942–954 (2020)
33. Tian, Q., Han, D., Li, K.C., Liu, X., Duan, L., Castiglione, A.: An intrusion detection approach based on improved deep belief network. Applied Intelligence 50(10), 3162–3178 (oct 2020)
34. Xiao, T., Han, D., He, J., Li, K.C., de Mello, R.F.: Multi-keyword ranked search based on mapping set matching in cloud ciphertext storage system. Connection Science 33(1), 95–112 (2021)
35. Xu, Z., Liang, W., Li, K.C., Xu, J., Zomaya, A.Y., Zhang, J.: A time-sensitive token-based anonymous authentication and dynamic group key agreement scheme for industry 5.0. IEEE Transactions on Industrial Informatics pp. 1–1 (2021)
36. Yu, Y., Li, Y., Tian, J., Liu, J.: Blockchain-based solutions to security and privacy issues in the internet of things. IEEE Wireless Communications 25(6), 12–18 (2018)
37. Zhang, Y., Sun, W., Xie, C.: Blockchain in smart city development—the knowledge governance framework in dynamic alliance. In: International Conference on Smart City and Intelligent Building. pp. 137–152. Springer (2018)
38. Zhang, Y., Kasahara, S., Shen, Y., Jiang, X., Wan, J.: Smart contract-based access control for the internet of things. IEEE Internet of Things Journal 6(2), 1594–1605 (2018)
39. Zraková, D., Demjanoviová, M., Kubina, M.: Online reputation in the transport and logistics field. Transportation Research Procedia 40, 1231–1237 (2019)

Manjie Zhai is currently pursuing the M.S.degree with the School of Information Engineering, Shanghai Maritime University, Pudong, China. Her current research interests include blockchain and internet of things security.

Dezhi Han received the B.S. degree in applied physics from the Hefei University of Technology, Hefei, China, in 1990, and the M.S. and Ph.D. degrees in computing science from the Huazhong University of Science and Technology, Wuhan, China, in 2001 and 2005, respectively. He is currently a Professor with the Department of Computer, Shanghai Maritime University, Pudong, China, in 2010. His current research interests include cloud and outsourcing security, blockchain, wireless communication security, network, and information security.

Chin-Chen Chang received the Ph.D. degree in computer engineering from National Chiao Tung University, Hsinchu, Taiwan, in 1982, and the B.E. and M.E. degrees in applied mathematics, computer and decision sciences from National Tsinghua University, Hsinchu, Taiwan, in 1977 and 1979, respectively. He was with National Chung Cheng University, Minxiong, Taiwan. Currently, he is a Chair Professor with the Department of Information Engineering and Computer Science, Feng Chia University, Taichung, Taiwan, since 2005. His current research interests include database design, computer cryptography, image compression, and data structures.

Zhijie Sun is currently pursuing the M.S.degree with the School of Information Engineering, Shanghai Maritime University, Pudong, China. His current research interests include blockchain and internet of things security.

Received: December 20, 2021; Accepted: March 15, 2022.

Computer Science and Information Systems 19(2):957–978 https://doi.org/10.2298/CSIS2201180100

A Dockerized Big Data Architecture for Sports Analytics

Yavuz Melih Özgüven[1], Utku Gönener[2], and Süleyman Eken[3]

[1] Kocaeli University, Department of Computer Engineering
Izmit 41001, Turkey
yavuzozguven@hotmail.com
[2] Kocaeli University, Faculty of Sports Sciences
Izmit 41001, Turkey
utku.gonener@kocaeli.edu.tr
[3] Kocaeli University, Department of Information Systems Engineering
Izmit 41001, Turkey
suleyman.eken@kocaeli.edu.tr

Abstract. The big data revolution has had an impact on sports analytics as well. Many large corporations have begun to see the financial benefits of integrating sports analytics with big data. When we rely on central processing systems to aggregate and analyze large amounts of sport data from many sources, we compromise the accuracy and timeliness of the data. As a response to these issues, distributed systems come to the rescue, and the MapReduce paradigm holds promise for large-scale data analytics. We describe a big data architecture based on Docker containers with Apache Spark in this paper. We evaluate the architecture on four data-intensive case studies in sport analytics including structured analysis, streaming, machine learning approaches, and graph-based analysis.

Keywords: big data, sports analytics, containers, wearable devices, IoT, reproducible research.

1. Introduction

Decision making in sports based on the information acquired by observation has changed with technological advances. Sports analytics has been more popular in recent days [53]. Sports analytics has the concept of using sports data to create valuable statistics for analysis the need for proper data models is present [50]. There are different approaches on how to treat sports data in combination with data analytics to create statistics and other beneficial information. The big collection of sports data then benefits the analysis and decision making from the sports games [34]. Analytics is applied to conclude advantages in the exercising of sports. The conclusions need to be originated from established data analytics, according to mathematical models that the sports industry has evaluated and are used in some manner.

Since the data rate has gone up in the latest years the need for efficient big data analytics has become more and more important. The increasing smart devices being carried have made data rate explode, and with the increasing sensors and interactions in society some smart solutions need to be carried [35]. Not only the increasing devices has made an impact, but also the behavior of the users. A technology such as positioning generates a boosted amount of data, and there are many areas such as business data, image data

and industrial process data [47]. When we rely on central processing systems to aggregate and analyze large amounts of sports data from many sources, we compromise the accuracy and timeliness of the data. As a result, we must apply distributed and parallel computing technology to sports analytics study.

Contributions to the literature with the paper can be listed as follows:

- Current MapReduce based frameworks offer poor support for reusing existing processing tools in sports data analytics pipelines. We give an open source architecture that introduces support for Docker containers in Apache Spark.
- We illustrate how to apply the architecture in four data-intensive applications in sport analytics, including structured analysis, streaming, machine learning approaches, and graph-based analysis.

The remainder of this article is organized as follows. Section 2 gives a literature review on structured sports data analysis, sport data streaming, machine learning approaches in sports, and graph-based sport data analysis. Section 3 gives a sports data search mechanism and repository analysis from a reproducible research perspective. Section 4 gives details of the containerized big data architecture. Section 5 presents the performance of the system. Section 6 summarizes and and also gives lessons learned and future work.

2. Related Works

This section will demystify the analytical thinking behind the data revolution in sports through a wide range of topics related to sport data analytics in the literature. We organize this section as four sub-sections.

2.1. Structured sport data analysis

This subsection summarizes structured sport data analysis in the literature. We can classify data analysis utilisation depending on the velocity and variation of data i.e. real-time, batch processing, and structured, semi-structured, unstructured. Analytics can acquire both insights and foresight from the data. An ELT process, extract-load-transform, where data is extracted and loaded in a raw format and transformation steps are diverted towards the database engine to be performed as small atomic tasks through SQL statements. The transformed data is then moved into a data model that is accessible by users. Following paragraph consists of structured sport data analysis related works.

Metulini [37] concerned with basketball data processing, and aimed to suggest an ad-hoc procedure to automatically filter a data matrix containing players' movement information to the moments in which the game is active, and by dividing the game into sorted and labelled actions as offensive or defensive. Knobbe et al. [29] worked on professional speed skating and devised a number of features that capture various aspects of sports events by aggregating discrete sequences of such events. The aggregation can be done in two ways: one that is easy to compute and interpret (uniform window), one that is more physiologically plausible but harder to compute (the Fitness-Fatigue model). SQL statements were used to perform these aggregations. Pers et al. [40] also used standard SQL to analyze large volumes of annotated sport motion data. Their goal was to detect particular types of play, activities, and predefined situations automatically, as well as generate numerous relevant information.

2.2. Sport data streaming

This subsection summarizes stream based sport data analysis in the literature. A number of low-cost wearable devices and gadgets aimed at sport tracking and monitoring have been launched to the market in recent years. Many major technology trends of other Internet of Things (IoT) solutions are shared by sport tracking and monitoring systems. Due to latency and bandwidth limits, cloud and fog computing principles may be a solution to the challenge of real-time analysis and feedback of these IoT devices.

Pustišek et al. [43] touched on the relevance of technology in sports for motor learning, as well as the features and limits of various sensors utilized for activity signal gathering, communication channels, and ways of communication. They created feedback systems that may be used for a variety of augmented motor learning applications using smart sports equipment. Grün et al. [21] created a system that can track a huge number of high-dynamic objects in real-time inside a pre-defined region of interest, such as during a football game. Probst et al. [42] designed a complete team sports analysis infrastructure. This system can identify collaborative events automatically, create statistics based on a continuous stream of raw locations, show the analysis findings in real time, and then save the analysis results in permanent storage for offline use and intuitive sketch-based video retrieval later. Capobianco et al. [7] proposed a formal methodology for designing an expert system based on big data acquired from various sources, the purpose of the system is to support real-time decisions for notational analysis in a sports environment. Haiyun and Yizhe [22] developed an integrated and extensive learning based Hadoop platform for forecasting game outcomes. Dinesh et al. [44] presented a system for detecting violence in a football stadium in real time. In the Spark environment, the HOG function was utilized to extract features from video frames. Proposed system alerts the security forces. Baerg [2] analyzed the athlete's performance with big data. Stein et al. [55] first discussed how to evaluate team sport data in general before proposing a multi-faceted approach that included pattern recognition, context-aware analysis, and visual explanation. Luo et al. [33] reported a wood-based triboelectric nanogenerator that is flexible and robust for self-powered sensing in sports big data analytics.

2.3. Machine learning based sport data analysis

This subsection summarizes machine learning based sport data analysis in the literature. Podgorelec et al. [41] created a new image dataset of four comparable sports (American football, rugby, soccer, and field hockey) and used CNN transfer learning with Hyper-Parameter Optimization (HPO) to categorize the images. Their proposed method was then compared to a conventional CNN and a CNN with transfer learning but handpicked hyper-parameters for fine-tuning. Constantinou et al. [11] developed probabilistic models based on possession rates and other historical statistics of various teams to predict the outcome of matches. Kapadia et al. [25] used machine learning techniques to solve the same problem but for the cricket world in the Indian Premier League (IPL). Jayalath [24] considered the popular logistic regression model to study the significance of one-day international (ODI) cricket predictors. Kerr [28] presented three experiments in his thesis. In the first experiment, three models were created utilizing various attributes to predict which side would win a particular game without knowing the score. Several classifiers were employed in the second experiment to predict which team created the sequence of

ball-events that happened during a game. And in the last one, he predicted which team attempted a given set of passes. Brooks et al. [5] focused on examining characteristics of passing in soccer and introduced two methods for obtaining insights from that. Ehrlich and Ghimire [13] took note of the effect the presence or absence of fans can have on a team's performance in Major League Baseball. He analyzed various scenarios in the context of physical distancing due to COVID-19 and used logit regression and a neural network to simulate the 2020 season.

Ghimire et al. [17] used Adjusted Plus-Minus (APM) measures to evaluate player contribution in basketball and hockey. APM measures estimate the impact of an individual player on his team's scores using seasonal play-by-play data. They used a two-stage least square (2-SLS) approach to test the robustness of a series of linear fixed effects regression models to explain variance in Real Plus-Minus (RPM) between player seasons. In order to identify essential features and generate interpretable models for sport data analytics in professional speed skating, Knobbe et al. [29] employed linear modeling and subgroup discovery. Vinué and Epifanio [56] developed a useful mathematical tool based on archetypoid analysis (ADA) to evaluate the worth of players and clubs in a league by analyzing their performance. In three cases, the value of archetypoids in sports was demonstrated using data from basketball and soccer. Janetzko et al. [48] created a method that allows users to interactively examine and evaluate movement characteristics and game events in high-frequency position-based soccer data at various degrees of detail. Sidle and Tran [52] applied multi-class classification methods to the problem of predicting baseball live pitch types. While Chu and Swartz [9] proposed a Bayesian inference system with parametric models to analyze fouling time distributions. Karetnikov [27] proposed a principally new complex performance prediction framework for cycling with are the Maximum Mean Power (MMPs) and the race position performance metrics.

2.4. Graph-based sport data analysis

This subsection summarizes graph based sport data analysis in the literature. Duch et al. [12] and Pena and Touchette [38] examined weighted pass graphs. Players were represented by nodes, passes by edges, and the efficiency of passes by weights. Cintia et al. [10] analyzed a passing network using network centrality metrics from two perspectives: passes between players and passes between pitch zones. Zheng et al. [61] predicted game outcomes from available sports statistics using a graph signal processing (GSP) perspective. Roane et al. [46] developed an approach to sports rankings that reflects the strength of each team while accounting for game results. They represented teams and the games between them as a digraph and considered minimizing the number of backedges in a ranking. Brandt and Brefeld [4] presented a graph-based approach to analyze player interaction in team sports. Shi and Tian [51] used a game graph from the perspective of Bayesian correction with game results to build a generalized PageRank model for sports. Wu et al. [58] created a social network from player positions and passings to comprehensively measure the importance of playing positions. Features such as degree, closeness, betweenness, eigenvector, and load centralities, as well as reciprocity, and clustering were used. Football Passing Networks[4] is a web application that allows users to engage with data visualizations of soccer passing networks.

[4] https://grafos-da-bola.netlify.app/

Although there are already approaches focused on different aspects of sports, to the best of our knowledge, there is no open source containerized big data architecture yet that jointly supports the structured-based, stream-based, machine learning-based, and graph-based sports data analytics. All of these topics are the most used types of data analysis in other big data fields.

Every day, developers find new ways to put containerization to work to solve their challenges. There is no shortage of ways to use containerization, and every application will likely produce unique benefits. Here are some of the most common benefits of our proposed containerized big data architecture: (i) Portability between different platforms and clouds—it's truly write once, run anywhere. (ii) Efficiency through using far fewer resources than VMs and delivering higher utilization of compute resources. (iii) Agility that allows developers to integrate with their existing DevOps environment. (iv) Improved security by isolating applications from the host system and from each other. (v) Faster app start-up and easier scaling.

3. Sport Dataset Search and Repository Analysis

This section covers searching a problem-specific dataset and repository analysis related to sport data analytics.

3.1. Sports Dataset Search

Many types of datasets exist in sports such as (i) raw dataset: game box scores; play-by-play; player tracking, (ii) extracted events: hits, runs, points, rebounds, assists, etc, and (iii) stats: batting avg, total bases, RBI, shooting %, etc. In general, it is very difficult to find a public dataset for a problem, and it is a problem that anyone cannot predict how much the dataset she/he find will work. Briefly, dataset sources can be summarized as follows: (i) websites: -leagues: MLB.com, NBA.com, -general: ESPN, baseball/basketball/football reference, FanGraphs, (ii) API/published: PitchF/X, Statcast, NBA Stats, (iii) curated (not necessarily free): Lahman Database, Retrosheet, armchairanalysis.com (cheap with .edu email), and (iv) other: -API tools and scrapers published on GitHub (lots of repos out there), -data collectives: Kaggle, data.world.

When seeking high-quality datasets, there are a few things to consider:

- The dataset should not be messy, otherwise significant time will be wasted on cleaning it. The cleaner, the better.
- The dataset should not have too many rows or columns, so it is easy to work with.
- There should be a question/decision to answer using the data.

Anyone can find a public dataset related to different sport branchs using well-known repositories such as Google Dataset Search, Kaggle, UCI Machine Learning Repository, and Data.gov. Also, there are different specific data sources such as StatsBomb Open Data[5], open football[6] for soccer, NFLsavant.com[7] for Amercan football, Lahman's Base-

[5] https://statsbomb.com/academy/
[6] https://openfootball.github.io/
[7] http://nflsavant.com/

ball Database[8] for baseball, FiveThirtyEight[9] for others, Sport Database [49] for cardiorespiratory data, Heimdallr [45] for action recognition and pose estimation and etc.

3.2. Repository Analysis

GitHub[10], a hosting platform for open-source software projects, has gained much popularity in recent years [31]. In contrast with competitors (e.g., SourceForge[11], Assembla[12]), Github offers more than just version control hosting, but also an easy-to-use and cheap or free (depending on the version) online tool for collaborative software development and other attractive features [20].

We consider sports analytics repositories and their data on GitHub to follow their growth and development processes. We use git command-line search CLI[13] to retrieve git repository "statistics". It provides a cli for searching github.com and supports repositories, code, issues and commits. These "statistics" include repos, code, commits, issues, users, wikis, and topics. Table 1 shows statistics for "sport analytics" keyword. According to these statistics, there are 297 repos which titles include "sport analytics". Mostly used three languages in repos are Jupyter Notebook, Python, and R. These are also mostly used languages in other data analysis/analytics works [14]. Similarly, other keywords related to sport such as "sport", "sport data", "sport materials", and "sport activity" can be searched.

Table 1. GitHub statistics for "sport analytics" keyword

Repositories	297 repos
Language	Jupyter Notebook (67), Python (45), R (44), HTML (30), Java (9), JavaScript (7), PHP (3), MATLAB (2), C (1), C++ (1)
Commints	58 commits
Issues	16 issues States (8 Closed and 8 Open) Languages (Python (5), Java (4), JavaScript (2), HTML (1), Jupyter Notebook (1), R (1))
Topics	# sports-analytics (121 repositories) # sport-analytics (7 repositories)
Wikis	7 wiki results
Users	8 users

Repositories also serve reproducibility. Reproducibility is the minimum attainable standard for assessing scientific claims. To fulfill this, researchers are required to make both their data and computer code available to their peers. This, however, still falls short of full replication since independently collected data is not used. Nevertheless, this standard allows an assessment to some degree by verifying the original data and codes [39][15].

[8] http://www.seanlahman.com/baseball-archive/
[9] https://data.fivethirtyeight.com/
[10] https://github.com/
[11] https://sourceforge.net/
[12] https://www.assembla.com/
[13] https://github.com/feinoujc/gh-search-cli

4. Containerized Architecture

This section gives the players of the our containerized big data architecture: Apache Spark and Docker.

4.1. Apache Spark

The amount of data being processed when streaming sports data, especially with multiple users and when streaming a broad set of activities, commands large amounts of computing power that cannot be provided by solely scaling up, meaning increasing the performance of a single machine. Instead, the performance required is achieved by scaling out, meaning distributing the computation across multiple machines [57]. Spark manages this scaling out by abstracting these machines as so-called execution nodes (worker nodes, slave nodes), on which programs (tasks), called sparkjobs, are run. These abstract execution nodes can also be separate processes on a single machine, efficiently utilizing multiple cores. Apache Spark can run in stand-alone settings, as well as on some popular platforms (e.g., Kubernetes [30], Mesos [23], and Hadoop YARN).

The distribution of tasks to these nodes, and the collection of results from them, is managed by the master node (driver node). It utilizes a HDFS (Apache Hadoop Distributed File System) to persist data across these nodes [60]. An illustration of this architecture can be seen in Fig. 1.

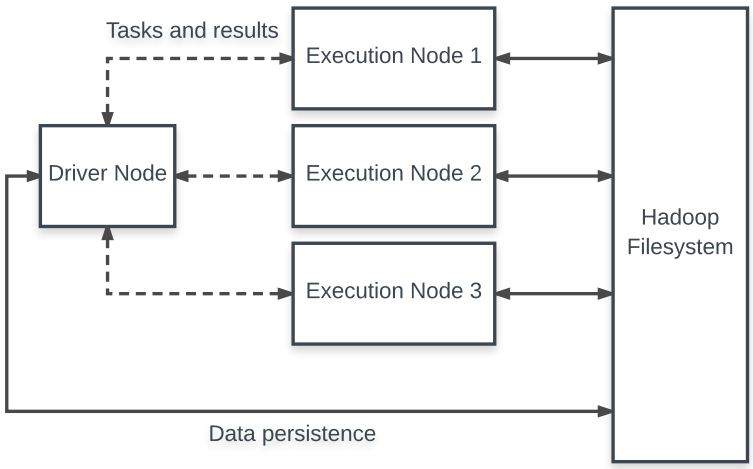


Fig. 1. Simplified diagram of a typical Spark cluster

Spark also offers functionality to perform machine learning, graph processing, structured data analysis and more on data from streams, files or databases, in a distributed setting, with just a few lines of code [26]. This means that Spark unifies and simplifies a lot of tasks in one framework that previously required multiple technologies. This has lead to a widespread adoption of Spark since its release in 2010, making it the biggest

open source big data project [60], with over 1600 contributors [18] and over 1000 adopting organizations. To enable efficient implementation of big data tasks, Spark introduces a concept called RDDs (Resilient Distributed Datasets), through which the parallelization, distribution and persistence of data is abstracted for the developer [16], see Fig. 2 for a simple example.

```
data = [1, 2, 3, 4, 5]
# Wrap the data into a RDD, which is distributed across
# execution nodes by Spark, ready for parallel processing
# sc is the so-called streaming context,
# which provides an interface with the cluster
distributedData = sc.parallelize(data)
# Add a map-action that increments each value,
# distributed across execution nodes
incrementedData = distributedData.map(lambda a: a + 1)
# Run a reduce-transformation, which is run on the driver node
# The RDD is automatically collected (persisted) to the driver node
incrementedData.reduce(lambda a, b: a + b)
# returns 20
```

Fig. 2. A simple code example showing how spark distributes data and collects the result back to the driver node

Furthermore, Spark is developed by the Apache Software Foundation, as is Kafka, which means they are designed to work well with each other. For example, the Python API of Spark offers a range of utility functions to build sparkjobs to consume a Kafka-stream very easily, which means constructing a sparkjob to act as a consumer for a Kafka-stream in a distributed setting can be achieved with very few lines of code, as can be seen in the wordcount-example in Fig. 3.

```
# Stream the data in 1-second windows
streaming_context = StreamingContext(sc, 1)  # 1 second window
# Connect to a kafka stream, specifying which kafka-topics to consume.
# See section "Kafka" for an explanation of topics.
stream = KafkaUtils.createStream(streaming_context,
                                 'docker:2181',
                                 "stream-1",
                                 {"topic-1": 1})
# Each window, count each word in each line
counts = stream.flatMap(lambda line: line.split(" ")) \
        .map(lambda word: (word, 1)) \
        .reduceByKey(lambda a, b: a + b)
```

Fig. 3. Consuming data from a stream and processing them in batches

Spark also allows for more complex functions to be applied to the data. The previous examples shown in Figs. 2 and 3 exclusively used lambda expressions. However, more

complex functions cannot be sensibly realized as lambda expressions, as they are by definition limited to a single expression. Also, developers might need to use variables defined outside of the function because their initialization is computationally intensive, or uses data that is only available on the driver node. One real strength and important characteristic of Spark is that it can transmit the whole closure of a sparkjob to the execution nodes, as long as they are serializable. This means that computationally expensive initialization of variables only need to be done once instead of on every execution node. Furthermore, imported packages such as libraries and frameworks are transmitted as well, which simplifies their usage in sparkjobs.

4.2. Docker

A container image is a packaged light-weight piece of software including, within itself, everything required to run correctly: code, run-time, system tools, system libraries, and settings. A container isolates software from its surroundings and will always run the same way regardless of the operating system or environment (e.g. development and staging). Make it possible to densely pack multiple apps on the same infrastructure. And help reduce conflicts between teams running different software on the same infrastructure [3], or running the same software on different machines. From Fig. 4 it is seen that in one single host there are three containers running. Each container contains the necessary environment variable inside. So, it is not necessary to have the all environment variable before in a host to run the application. Container itself will create the environment to run the application.

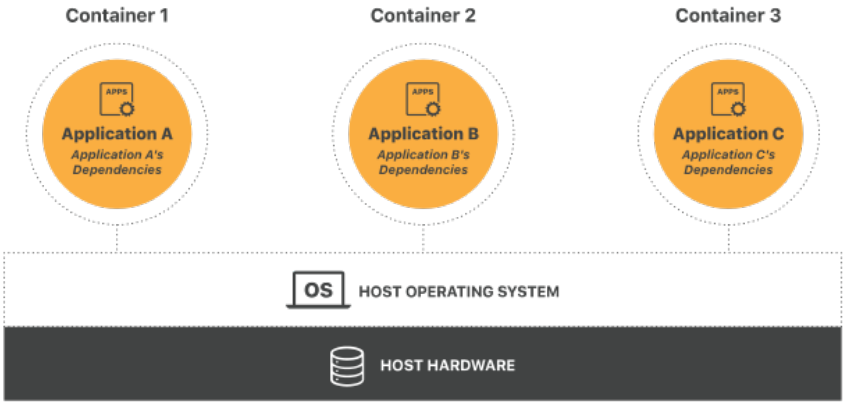


Fig. 4. Container architecture

Docker is one of the most popular software containerization platforms. Developers, operators, and enterprises use it for the previously mentioned merits. Users may substantially shorten the time between developing code and executing it in production by utilizing Docker's techniques for shipping, testing, and deploying code rapidly. Because of the isolation and security, users may operate several containers on a single host [1].

5. Experimental Results and Discussion

5.1. Experimental Setup

All performance tests are done on Microsoft Azure. Specifications of used server are as following: Zone: East-US, Cpu: 2 core, Memory: 8 GB, OS: Ubuntu 16.04-LTS, Disk: 30 GB. Package dependencies are as following: spark-core_2.12, spark-sql_2.12, spark-mllib_2.12, isolation-forest_3.0.0_2.12, spark-graphx_2.10, pulsar-client 2.6.2, and pulsar-spark 2.6.2. Also, we use an open source programming library, MaRe [8]. It enables scalable data-intensive processing.

5.2. Case Studies

Case study 1: Extracting interesting information about footballers with SQL statements Spark SQL is a module for managing structured data. With Spark SQL, it is feasible to query structured data by utilizing either structured query language or a similar API. It can be used with Python, R, and similar languages. It ensures uniform data access. SQL and DataFrames supply a common way to connect to various data sources, including JDBC, Hive, JSON, Parquet, etc. Spark SQL can scale up to hundreds of nodes simultaneously by utilizing the Spark framework.

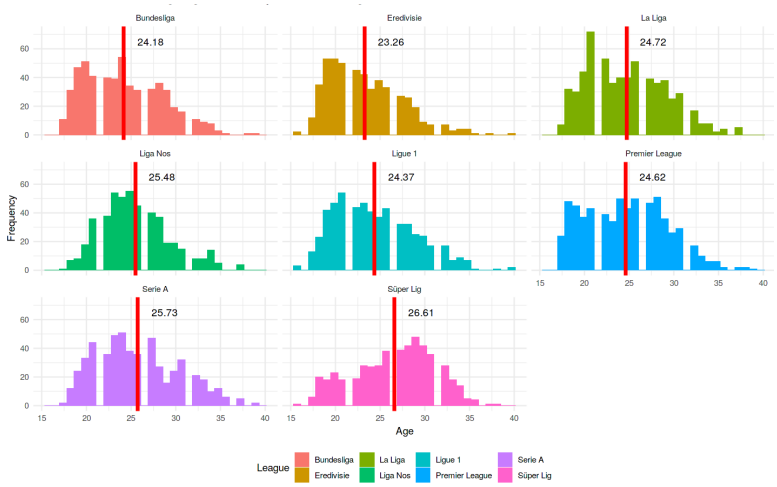


Fig. 5. Distribution and the average age of the players in each league for FIFA 19

In Apache Spark, there are various abstractions for data: Resilient Distributed Datasets (RDDs), DataFrames, Datasets, and SQL Tables. All of these various abstractions show distributed collections of data. RDD was the main API in Spark since its beginning. RDD is an unchangeable distributed compilation of data. RDD is split over nodes in the cluster and might be used simultaneously. From the Apache Spark version 2.0 onwards, DataFrames have been the principal API in Spark. DataFrame's syntax is more instinctive

than of RDD's, but their functionality doesn't differ. RDDs are part of the low-level API and the DataFrames are part of the Structured APIs. Similar to an RDD, a DataFrame is an unchangeable distributed compilation of data. Different from an RDD, in a DataFrame, data is formed into named columns. The RDD functionally and visually looks like to the Pandas in Python and R DataFrames. It is also comparable to an Excel Spreadsheet. It is possible to use them to manipulate, explore, and import the data. Additionally, SQL queries might be used within Spark syntax.

FIFA 20 complete player dataset[14] is used for this case study. Players' data for Career Mode from FIFA 15 to FIFA 20 is included in the databases. The information enables for numerous comparisons of the same players over the videogame's past six versions. Fig. 5 shows distribution and the average age of the players in each league for FIFA 19.

Some questions and their SQL statements on "FIFA 20 complete player dataset" is as following:

- Top 10 country with highest mean wage

```
SELECT nationality, AVG(wage_eur), AVG(overall) FROM fifa
GROUP BY nationality ORDER BY AVG(wage_eur) DESC limit 10
```

- Age vs overall rating vs wage

```
SELECT age, AVG(wage_eur), AVG(overall) FROM fifa
GROUP BY age ORDER BY AVG(overall) DESC
```

- Club vs potential top 10

```
SELECT club, AVG(potential) FROM fifa
GROUP BY club ORDER BY AVG(potential) DESC limit 10
```

- Weak foot count

```
SELECT weak_foot, Count(weak_foot) FROM fifa
GROUP BY weak_foot ORDER BY weak_foot ASC
```

More questions and their statements are available at GitHub repo. Putting these queries into jar (sql.jar) and then copying to an image containing java on the docker named 'sql', the following code snippet in Fig. 6 is run. We initialize MaRe by passing it a player dataset that was previously loaded as an RDD (rddPlayer). We implement the SQL statements' run using the map primitive. We set input and output mount points as text files then we specify a Docker image as sql. Finally, we specify the sql command. As seen, existing other serial tools can be run in MapReduce fashion.

Case Study 2: Machine learning practices on different sport datasets with Spark MLlib A powerful analytics library and Spark MLlib [36], a built-in general-purpose machine learning framework, are the key features contributing to the use of Spark. Because of its simplicity, language compatibility, scalability, performance, and ease of interaction with other tools, it is highly popular among data scientists. It allows data scientists to focus entirely on data-related activities, bypassing the complexity of infrastructure and

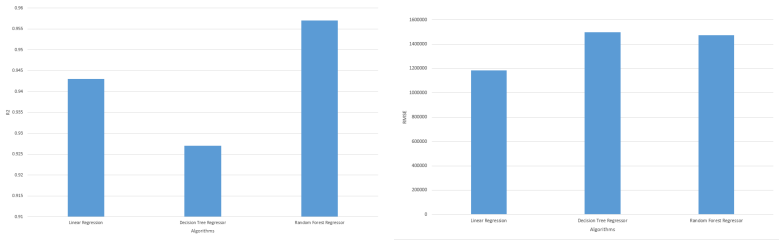
[14] https://www.kaggle.com/stefanoleone992/fifa-20-complete-player-dataset

```
val rddPlayer = sc.textFile(path="players_20.csv")
val res = new MaRe(rddPlayer)
    .map(
        inputMountPoint = TextFile("\input"),
        outputMountPoint = TextFile("\output"),
        imageName = "sql",
        command = "java -jar sql.jar > out")
    .rddPlayer.collect()
res.foreach(println(_))
```

Fig. 6. Virtual screening of structured analysis in MaRe

setup. Spark MLlib includes a set of efficient machine learning methods (such as regression, classification, clustering, filtering, and collaboration) as well as the ability to adapt the algorithms for specific use cases.

We concern regression, clustering, and classification on different sport datasets. Same FIFA 20 complete player dataset is used for regression purpose. Regression process includes following sub-steps: (i) separating features into categorical and numeric ones, (ii) converting categorical features to numeric values with StringIndexer, (iii) merging dataframes, (iv) vectorizing these merged features, (v) setting training and testing data, (vi) testimonial estimation with different regression models such as linear regressor, decision tree regressor, and random forest regressor, and (vi) measuring their performances with R^2 and Root Mean Square Error (RMSE) metrics. Fig. 7 shows performance results.



(a) R^2 **(b)** RMSE

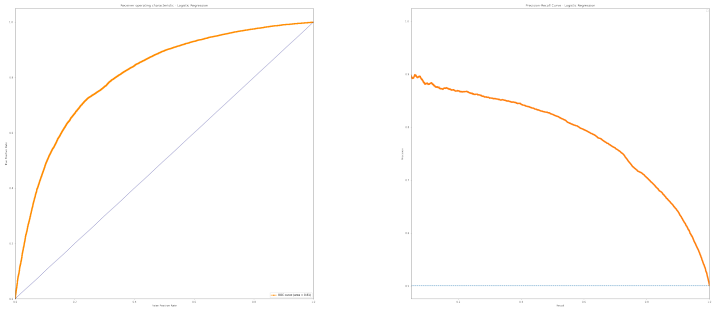
Fig. 7. Performance results for different regression algorithms on FIFA 20 complete player dataset

Association of Tennis Professionals (ATP) Matches dataset[15] is used for classification task. Individual csv files for ATP tournaments from 2000 to 2017 may be found in these databases. Fig. 8 shows player's performance of their careers. We concern binary classification (prediction whether a player will beat the match or not) problem here. Classification process includes following sub-steps: (i) converting categorical features to numeric values with StringIndexer, (ii) target label assigning as 0 or 1, (iii) vectorizing features, (iv) setting training and testing data, (v) fitting different classification models

[15] https://www.kaggle.com/gmadevs/atp-matches-dataset

such as logistic regression, decision tree classifier, and random forest classifier, and (vi) measuring their performances with precision, recall, F1-score metrics. Fig. 9 shows area under precision-recall and ROC curves for Logistic regression model. Table 2 shows classification performance results on ATP Matches dataset.

Fig. 8. Grand slam match wins per year



(a) area under precision-recall curve **(b)** ROC curve

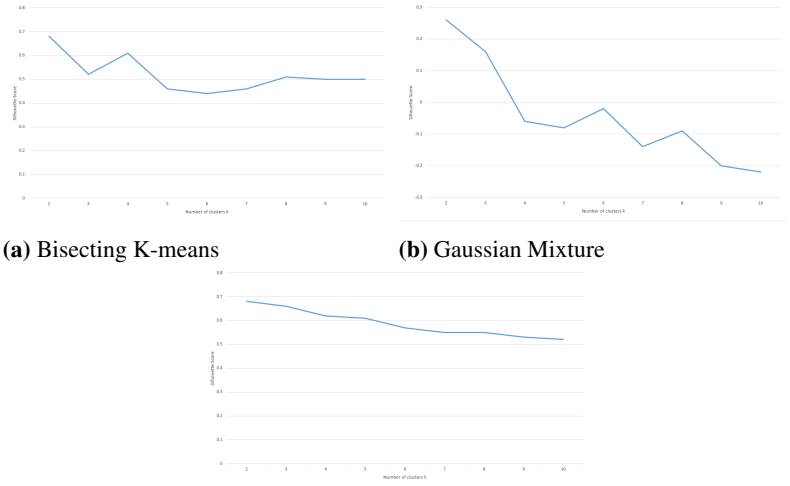
Fig. 9. Performance results for logistic regression

In clustering task, it is aimed to group goalkeepers with similar characteristics by using FIFA 20 complete player dataset. The players in the goalkeeper position are grouped

Table 2. Performance results for different classification algorithms on ATP Matches dataset

Classification approach	Precision	Recall	F1-score
Logistic regression	0.63	0.91	0.73
Decision tree classifier	0.66	0.84	0.71
Random forest classifier	0.62	0.84	0.69

by using the average of the goalkeeper characteristics and the average of overall and potential properties. Silhouette method is used to choose the best k-value. Fig. 10 shows the best k-values for different methods.



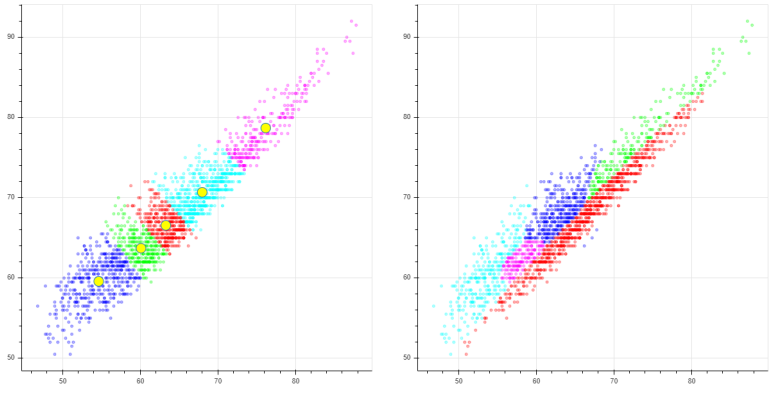
(a) Bisecting K-means **(b)** Gaussian Mixture

(c) K-means

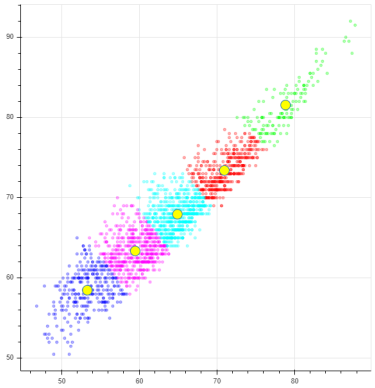
Fig. 10. Determining the optimal number of clusters for different algorithms

After determining the best k-value, following steps are done: (i) fitting different clustering models such as Bisecting K-means, Gaussian Mixture, and K-means, and (ii) visual results for these algorithms. Fig. 11 shows clustering results on FIFA 20 complete player dataset. Fig. 12 virtual screening of classification task in MaRe. Other tasks such as regression and clustering are realized in same way.

Case Study 3: Anomaly detection in multimodal eSports data using Spark Streaming and Apache Pulsar For more sophisticated data processing, Apache Spark is utilized in conjunction with Hadoop. Resilient Distributed Datasets is an efficient in-memory (RAM) cluster computing data format included with Spark Engine (RDDs). The system's data aggregation is done by Spark Streaming, which can handle both online and offline data streams.



(a) Bisecting K-means **(b)** Gaussian Mixture



(c) K-means

Fig. 11. Clustering results for different algorithms on FIFA 20 complete player dataset

```
val rddCluster = sc.textFile(path="tennis.csv")
val res = new MaRe(rddCluster)
    .map(
        inputMountPoint = TextFile("\input"),
        outputMountPoint = TextFile("\output"),
        imageName = "mllib",
        command = "java -cp project.jar Classification > output")
    .rddCluster.collect()
res.foreach(println(_))
```

Fig. 12. Virtual screening of classification task in MaRe

In this paper, The high-performance distributed messaging platform Apache Pulsar[16] (version 2.6.2) is utilized for the topic-based pub/sub system. It was initially developed by Yahoo and is now part of the Apache Software Foundation. It's used for reporting,

[16] https://pulsar.apache.org/

monitoring, marketing and advertising, customization, and fraud detection, and it's utilized for gathering and processing diverse events in near-realtime. Pulsar has been used to enhance the user experience at eBay, for example, by monitoring user interactions and behaviors. In terms of features and use cases, Pulsar is quite similar to Apache Kafka. It has excellent scalability for large-scale message processing, with high throughput and low end-to-end latency. Messages received are continuously saved with Apache BookKeeper, and message transmission between producers and consumers is assured. While Pulsar is not a full-fledged stream processing framework like Apache Storm or Spark Streaming, it does offer some light stream processing capabilities via Pulsar Functions.

Electroencephalography (EEG) data in multimodal eSports dataset[17] is used for streaming task [54]. Sensor data is collected from 10 players in 22 matches in League of Legends. In this task, it is aimed to detect anomalies in the sensor data of e-sports players sent via Apache Pulsar during the tournament. The anomaly detection model is created by using all the features in the sensor data with the IsolationForest algorithm [32], and then the anomalies are detected with this model. Model building process includes following substeps: (i) feature selection, (ii) vectorizing features, (iii) setting training and testing data, (iv) fitting the IsolationForest model. Anomaly detection process in real-time includes following sub-steps: (i) creation Pulsar client, (ii) making Spark Streaming Pulsar Reciever configurations and loading IsolationForest model, (iii) converting data received in batch form into a string array, and (iv) converting batch into a vector and combining it in a dataframe and anomaly detection over the model. Fig. 13 virtual screening of streaming task in MaRe.

```
val rddStream = sc.textFile(path="esports.csv")
val res = new MaRe(rddStream)
    .map(
        inputMountPoint = TextFile("\input"),
        outputMountPoint = TextFile("\output"),
        imageName = "anomaly",
        command = "java -jar project.jar > output")
    .rddStream.collect()
res.foreach(println(_))
```

Fig. 13. Virtual screening of streaming task in MaRe

Case Study 4: Football passing networks using Spark GraphX Spark GraphX [59] extends RDD by introducing graphs and graph-parallel computation capabilities. It provides various graph manipulation operations and graph-based algorithms (i.e. triangle counting, counted components, PageRank). Once the analysis process is done and results are obtained, they can be visualized for better understanding.

StatsBomb Open Data[18] is used for graph-based sports data analysis task. The data is provided as JSON files exported from the StatsBomb Data API. Here, we use events

[17] https://github.com/smerdov/eSports_Sensors_Dataset
[18] https://github.com/statsbomb/open-data

data. Events for each match are stored in events as json documents.In this section, we focus on football passing networks using Spark GraphX. The passing networks are based on a (generally basic) approach to the graphs theory or analysis, where it is considered the existence of: 1) individual entities (nodes or vertices) which belong to a population or specific group, and 2) the connections between them (edges) in terms an interaction to measure. So, if we translate this to football, the nodes are the players of a same team and the edges are the passes between them [19][6].

Passing networks are created as following: (i) creating nodes from player who do the pass and player who receive the pass, (ii) creating edges from nodes in a pass relationship, and (iii) graph construction using nodes and edges. When we define the data visualization mapping these are the most frequent considerations: (i) Nodes position- Mean player location when they do and/or receive a pass, (ii) nodes size- variable size depending on amount of passes, (iii) edges color- colored by amount of passes between specific two nodes (0-9, 10-19, 20-29, 30+), (iv) edges direction- this detail is ommited, (v) player ID- text (surname) close to them. Fig. 14 shows the Barcelona's passing network against Deportivo. Fig. 15 virtual screening of graph-based analysis task in MaRe.

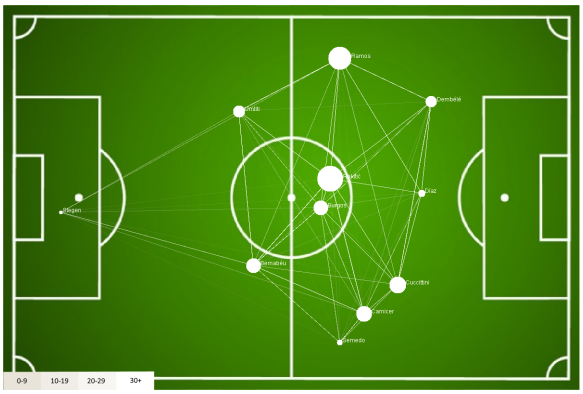


Fig. 14. Barcelona's passing network against Deportivo

```
val rddGraph = sc.textFile(path="statsbomb_event.csv")
val avg_pos = new MaRe(rddGraph).map(...).rdd
val netw = new MaRe(rddGraph).map(...).rdd
val avg_list = avg_pos.map {...}.collect().toList
val netw = netw.map {...}.collect().toList

Draw.network = netw_list
Draw.data = avg_list
```

Fig. 15. Virtual screening of graph-based analysis task in MaRe

6. Conclusions

The key contributions and findings of this work are summarized in this section. It also includes some broad "lessons learned" from the perspectives of both sports data analytics and big data, as well as potential future research topics.

6.1. Summary

The aim of this paper was to show how to analyze different sports data using many approaches from the research field of big data and distributed systems. To that purpose, we identified a number of flaws in the current literature and contributed to two major areas of sports analytics: (1) For sports data analytics pipelines, we offer a big data architecture based on Docker containers in Apache Spark. We presented an open source architecture that adds Docker container functionality to Apache Spark. (2) We presented the architecture in four data-intensive case studies in sports analytics, including structured analysis, streaming, machine learning techniques, and graph-based analysis.

6.2. Lessons learned

We looked at how the study disciplines of sports data analytics, distributed systems, and big data may be integrated, as well as what these research topics have to offer each other, in this part. We compiled a list of general findings and recommendations for sports data analytics practitioners and big data researchers. **Academic awareness:** It has been suggested that there is lot of doubt in the world of sports about the real value of business intelligence and analytics tools. The sports analytics utilisation level and practice is relatively neglected in the academic literature. This paper tested and addressed these ideas and contributed to the existing academic literature. **Reproducible research:** Reproducibility is the minimum attainable standard for assessing scientific claims. Researchers must make both their data and computer code open to their colleagues in order to do this. This, however, still falls short of full replication since independently collected data is not used. Nevertheless, this standard allows an assessment to some degree by verifying the original data and codes. Also, repository analysis and data search are important mechanisms in the context of reproducibility. **Containerized sports data processing:** Users may substantially shorten the time between developing code and executing it in production by utilizing Docker's techniques for shipping, testing, and deploying code rapidly. **Different types of data analytics:** We presented the architecture in four data-intensive case studies in sport analytics, including structured analysis, streaming, machine learning approaches, and graph-based analysis. **Big data:** The fields of big data and artificial intelligence provide a variety of approaches for extracting information, knowledge, wisdom, and judgment from raw data, which may be utilized to answer critical issues in sports analytics.

From the view of big data and distributed systems, we explore five lessons learned. **Domain knowledge:** Domain expertise might help big data and artificial intelligence models perform better. **Interpretability:** Experts are interested in putting the findings of analytics into effect in the end. It is to communicate these findings aesthetically, such as (interactive) drawings, graphs, and maps, to make this easier. **Ground truth data acquisition:** Real-world data in various sports sometimes lacks ground truth labeling. These may be difficult to get by or just do not exist.

6.3. Future work

Containerized sports data analytics is the paper's contribution. However, there are several unanswered issues and problems in the subject. This section outlines a number of potential future research directions. (i) Data privacy is a chief concern (buying fan data and types of data and how data is analysed). They apply to a wide range of sectors and are not limited to sports. As a result, it's a huge work to refine this data so that it's fit for fan consumption. (ii)In sports, data analytics is critical. As previously said, accurate data is critical in aiding in the improvement of stadium services. (iii) A sport-specific platform that brings together rights holders, sponsors and other stakeholders can be created. (iv) Personalized sport agility training systems can be created using sports nutrition, exercise drills, player activities, tactics, techniques via big data analytics' capabilities.

Software. Code underlying this article are available in Github, at https://github.com/yavuzozguven/Dockerized-Sport-Data-Analytics.

References

1. Anderson, C.: Docker. IEEE Software 32(3), 102–105 (2015)
2. Baerg, A.: Big data, sport, and the digital divide: Theorizing how athletes might respond to big data monitoring. Journal of Sport and Social Issues 41(1), 3–20 (2017)
3. Boettiger, C.: An introduction to docker for reproducible research. ACM SIGOPS Operating Systems Review 49(1), 71–79 (2015)
4. Brandt, M., Brefeld, U.: Graph-based approaches for analyzing team interaction on the example of soccer. In: MLSA@ PKDD/ECML. pp. 10–17 (2015)
5. Brooks, J., Kerr, M., Guttag, J.: Using machine learning to draw inferences from pass location data in soccer. Statistical Analysis and Data Mining: The ASA Data Science Journal 9(5), 338–349 (2016)
6. Buldú, J.M., Busquets, J., Martínez, J.H., Herrera-Diestra, J.L., Echegoyen, I., Galeano, J., Luque, J.: Using network science to analyse football passing networks: Dynamics, space, time, and the multilayer nature of the game. Frontiers in psychology 9, 1900 (2018)
7. Capobianco, G., Di Giacomo, U., Mercaldo, F., Santone, A.: A formal methodology for notational analysis and real-time decision support in sport environment. In: 2018 IEEE International Conference on Big Data (Big Data). pp. 5305–5307. IEEE (2018)
8. Capuccini, M., Dahlö, M., Toor, S., Spjuth, O.: Mare: Processing big data with application containers on apache spark. GigaScience 9(5), giaa042 (2020)
9. Chu, D., Swartz, T.B.: Foul accumulation in the nba. Journal of Quantitative Analysis in Sports 1(ahead-of-print) (2020)
10. Cintia, P., Rinzivillo, S., Pappalardo, L.: A network-based approach to evaluate the performance of football teams. In: Machine learning and data mining for sports analytics workshop, Porto, Portugal (2015)
11. Constantinou, A.C., Fenton, N.E., Neil, M.: pi-football: A bayesian network model for forecasting association football match outcomes. Knowledge-Based Systems 36, 322–339 (2012)
12. Duch, J., Waitzman, J.S., Amaral, L.A.N.: Quantifying the performance of individual players in a team activity. PloS one 5(6), e10937 (2010)
13. Ehrlich, J., Ghimire, S.: Covid-19 countermeasures, major league baseball, and the home field advantage: Simulating the 2020 season using logit regression and a neural network. F1000Research 9(414), 414 (2020)
14. Eken, S.: An exploratory teaching program in big data analysis for undergraduate students. Journal of Ambient Intelligence and Humanized Computing 11(10), 4285–4304 (2020)

15. Eken, S., Şara, M., Satılmış, Y., Karslı, M., Tufan, M.F., Menhour, H., Sayar, A.: A reproducible educational plan to teach mini autonomous race car programming. The International Journal of Electrical Engineering & Education 57(4), 340–360 (2020)
16. Foundation, A.: Spark Overview. https://spark.apache.org/docs/latest/index.html (2021), accessed 21-February-2021
17. Ghimire, S., Ehrlich, J.A., Sanders, S.D.: Measuring individual worker output in a complementary team setting: Does regularized adjusted plus minus isolate individual nba player contributions? PloS one 15(8), e0237920 (2020)
18. GitHub: Apache Spark Contributors. https://github.com/apache/spark (2021), accessed 11-February-2021
19. Gonçalves, B., Coutinho, D., Santos, S., Lago-Penas, C., Jiménez, S., Sampaio, J.: Exploring team passing networks and player movement dynamics in youth association football. PloS one 12(1), e0171156 (2017)
20. Gousios, G.: The ghtorent dataset and tool suite. In: 2013 10th Working Conference on Mining Software Repositories (MSR). pp. 233–236. IEEE (2013)
21. von der Grün, T., Franke, N., Wolf, D., Witt, N., Eidloth, A.: A real-time tracking system for football match and training analysis. In: Microelectronic systems, pp. 199–212. Springer (2011)
22. Haiyun, Z., Yizhe, X.: Sports performance prediction model based on integrated learning algorithm and cloud computing hadoop platform. Microprocessors and Microsystems 79, 103322 (2020)
23. Hindman, B., Konwinski, A., Zaharia, M., Ghodsi, A., Joseph, A.D., Katz, R.H., Shenker, S., Stoica, I.: Mesos: A platform for fine-grained resource sharing in the data center. In: NSDI. vol. 11, pp. 22–22 (2011)
24. Jayalath, K.P.: A machine learning approach to analyze odi cricket predictors. Journal of Sports Analytics 4(1), 73–84 (2018)
25. Kapadia, K., Abdel-Jaber, H., Thabtah, F., Hadi, W.: Sport analytics for cricket game results using machine learning: An experimental study. Applied Computing and Informatics (2020)
26. Karau, H., Warren, R.: High performance Spark: best practices for scaling and optimizing Apache Spark. " O'Reilly Media, Inc." (2017)
27. Karetnikov, A.: Application of data-driven analytics on sport data from a professional bicycle racing team (2019)
28. Kerr, M.G.S.: Applying machine learning to event data in soccer. Ph.D. thesis, Massachusetts Institute of Technology (2015)
29. Knobbe, A., Orie, J., Hofman, N., van der Burgh, B., Cachucho, R.: Sports analytics for professional speed skating. Data Mining and Knowledge Discovery 31(6), 1872–1902 (2017)
30. Kubernetes:
31. Lima, A., Rossi, L., Musolesi, M.: Coding together at scale: Github as a collaborative social network. In: Proceedings of the International AAAI Conference on Web and Social Media. vol. 8 (2014)
32. Liu, F.T., Ting, K.M., Zhou, Z.H.: Isolation forest. In: 2008 eighth ieee international conference on data mining. pp. 413–422. IEEE (2008)
33. Luo, J., Wang, Z., Xu, L., Wang, A.C., Han, K., Jiang, T., Lai, Q., Bai, Y., Tang, W., Fan, F.R., et al.: Flexible and durable wood-based triboelectric nanogenerators for self-powered sensing in athletic big data analytics. Nature communications 10(1), 1–9 (2019)
34. Marr, B.: Big Data: Using SMART big data, analytics and metrics to make better decisions and improve performance. John Wiley & Sons (2015)
35. Mayer-Schönberger, V., Cukier, K.: Big data: A revolution that will transform how we live, work, and think. Houghton Mifflin Harcourt (2013)
36. Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., Liu, D., Freeman, J., Tsai, D., Amde, M., Owen, S., et al.: Mllib: Machine learning in apache spark. The Journal of Machine Learning Research 17(1), 1235–1241 (2016)

37. Metulini, R.: Filtering procedures for sensor data in basketball. arXiv preprint arXiv:1806.10412 (2018)
38. Pena, J.L., Touchette, H.: A network theory analysis of football strategies. arXiv preprint arXiv:1206.6904 (2012)
39. Peng, R.D.: Reproducible research in computational science. Science 334(6060), 1226–1227 (2011)
40. Pers, J., Kovacic, S., Vuckovic, G.: Analysis and pattern detection on large amounts of annotated sport motion data using standard sql. In: ISPA 2005. Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis, 2005. pp. 339–344. IEEE (2005)
41. Podgorelec, V., Pečnik, Š., Vrbančič, G.: Classification of similar sports images using convolutional neural network with hyper-parameter optimization. Applied Sciences 10(23), 8494 (2020)
42. Probst, L., Rauschenbach, F., Schuldt, H., Seidenschwarz, P., Rumo, M.: Integrated real-time data stream analysis and sketch-based video retrieval in team sports. In: 2018 IEEE International Conference on Big Data (Big Data). pp. 548–555. IEEE (2018)
43. Pustišek, M., Wei, Y., Sun, Y., Umek, A., Kos, A.: The role of technology for accelerated motor learning in sport. Personal and Ubiquitous Computing pp. 1–10 (2019)
44. R, D.J.S., Fenil, E., Manogaran, G., Vivekananda, G., Thanjaivadivel, T., Jeeva, S., Ahilan, A.: Real time violence detection framework for football stadium comprising of big data analysis and deep learning through bidirectional lstm. Computer Networks 151, 191–200 (2019)
45. Riegler, M., Dang-Nguyen, D.T., Winther, B., Griwodz, C., Pogorelov, K., Halvorsen, P.: Heimdallr: a dataset for sport analysis. In: Proceedings of the 7th International Conference on Multimedia Systems. pp. 1–6 (2016)
46. Roane, A.R., Ekkaewnumchai, C., McNamara, C.W., Richards, K.: Graph-based sports rankings. Tech. rep., Worcester Polytechnic Institute (2019)
47. Runkler, T.A.: Data Analytics. Springer (2020)
48. Sacha, D., Stein, M., Schreck, T., Keim, D.A., Deussen, O., et al.: Feature-driven visual analytics of soccer data. In: 2014 IEEE conference on visual analytics science and technology (VAST). pp. 13–22. IEEE (2014)
49. Sbrollini, A., Morettini, M., Maranesi, E., Marcantoni, I., Nasim, A., Bevilacqua, R., Riccardi, G.R., Burattini, L.: Sport database: Cardiorespiratory data acquired through wearable sensors while practicing sports. Data in brief 27, 104793 (2019)
50. Severini, T.A.: Analytic methods in sports: Using mathematics and statistics to understand data from baseball, football, basketball, and other sports. Crc Press (2020)
51. Shi, J., Tian, X.Y.: Learning to rank sports teams on a graph. Applied Sciences 10(17), 5833 (2020)
52. Sidle, G., Tran, H.: Using multi-class classification methods to predict baseball pitch types. Journal of Sports Analytics 4(1), 85–93 (2018)
53. Silva, R.M.: Sports analytics. Ph.D. thesis, Science: Statistics and Actuarial Science (2016)
54. Smerdov, A., Zhou, B., Lukowicz, P., Somov, A.: Collection and validation of psycophysiological data from professional and amateur players: a multimodal esports dataset. arXiv preprint arXiv:2011.00958 (2020)
55. Stein, M., Janetzko, H., Seebacher, D., Jäger, A., Nagel, M., Hölsch, J., Kosub, S., Schreck, T., Keim, D.A., Grossniklaus, M.: How to make sense of team sport data: From acquisition to data modeling and research aspects. Data 2(1), 2 (2017)
56. Vinué, G., Epifanio, I.: Archetypoid analysis for sports analytics. Data Mining and Knowledge Discovery 31(6), 1643–1677 (2017)
57. Wolke, A., Meixner, G.: TwoSpot: A Cloud Platform for Scaling Out Web Applications Dynamically, pp. 13–24. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
58. Wu, Y., Xia, Z., Wu, T., Yi, Q., Yu, R., Wang, J.: Characteristics and optimization of core local network: Big data analysis of football matches. Chaos, Solitons & Fractals 138, 110136 (2020)

59. Xin, R.S., Gonzalez, J.E., Franklin, M.J., Stoica, I.: Graphx: A resilient distributed graph system on spark. In: First international workshop on graph data management experiences and systems. pp. 1–6 (2013)
60. Zaharia, M., Xin, R.S., Wendell, P., Das, T., Armbrust, M., Dave, A., Meng, X., Rosen, J., Venkataraman, S., Franklin, M.J., et al.: Apache spark: a unified engine for big data processing. Communications of the ACM 59(11), 56–65 (2016)
61. Zheng, H., Cheung, G., Fang, L.: Analysis of sports statistics via graph-signal smoothness prior. In: 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA). pp. 1071–1076. IEEE (2015)

Yavuz Melih Güven received the BSc degree in Computer Engineering from Kocaeli University in 2021. He is now MSc candidate in Information Systems Engineering at Kocaeli University. He is interested in big data, distributed systems and machine learning.

Utku Gönener received the BSc degree in Electrical and Electronics Engineering from Okan University in 2012. He is now PhD candidate in Sports Sciences at Kocaeli University. Currently, he works as a research assistant at the same university. His interests are exercise sciences and periodization, biomechanics, physiological and performance tests.

Süleyman Eken received his MS degree and PhD degree in Computer Engineering at the Kocaeli University. He was a research assistant at Kocaeli University, Turkey, from 2010 to 2019. Currently, he works as an Associate Professor of Information Systems Engineering, Kocaeli University, Izmit, Turkey. His main research work focuses on distributed systems and big data analysis.

Received: January 18, 2022; Accepted: March 20, 2022.

Computer Science and Information Systems 19(2):979–1005 https://doi.org/10.2298/CSIS2202150110

A Novel Hybrid Recommender System Approach for Student Academic Advising Named COHRS, Supported by Case-based Reasoning and Ontology

Charbel Obeid[1], Christine Lahoud[2], Hicham El Khoury[3], and Pierre-Antoine Champin[4]

[1] LIRIS, Caude Bernard University Lyon 1
France
cobeid@liris.cnrs.fr
[2] French University in Egypt
Egypt
Christine.lahoud@ufe.edu.eg
[3] LaRRIS, Lebanese University
Lebanon
hkhoury@ul.edu.lb
[4] LIRIS, Caude Bernard University Lyon 1
France
Pierre-antoine.champin@univ-lyon1.fr

Abstract. The recent development of the World Wide Web, information, and communications technology have transformed the world and moved us into the data era resulting in an overload of data analysis. Students at high school use, most of the time, the internet as a tool to search for universities/colleges, university's majors, and career paths that match their interests. However, selecting higher education choices such as a university major is a massive decision for students leading them, to surf the internet for long periods in search of needed information. Therefore, the purpose of this study is to assist high school students through a hybrid recommender system (RS) that provides personalized recommendations related to their interests. To reach this purpose we proposed a novel hybrid RS approach named (COHRS) that incorporates the Knowledge base (KB) and Collaborative Filtering (CF) recommender techniques. This hybrid RS approach is supported by the Case-based Reasoning (CBR) system and Ontology. Hundreds of queries were processed by our hybrid RS approach. The experiments show the high accuracy of COHRS based on two criteria namely the "accuracy of retrieving the most similar cases" and the "accuracy of generating personalized recommendations". The evaluation results show the percentage of accuracy of COHRS based on many experiments as follows: 98 percent accuracy for "retrieving the most similar cases" and 95 percent accuracy for "generating personalized recommendations".

Keywords: Knowledge base, Collaborative Filtering, Hybrid Recommender System, Case-based Reasoning, Ontology.

1. Introduction

Academic advising at most high schools is limited in its ability to help students in identifying appropriate educational pathways. For example, choosing a university and a universtiy

major is a challenging task rife with anxiety that gets students confused. (Cuseo, J., 2003) (V.N., 2007) revealed that 20 to 50 percent of students in the United States start their university journey with an undecided major and 50 to 75 percent of learners in higher education have changed their major at least once before graduation. This suggests that students' career choices are unclear upon university admission and enrollment. Besides (Tett et al., 2017), (Siri et al., 2016) outlined the transition from high school to university by socio-cultural perspectives affected by personal factors and the learning environment, comprising students' previous experiences.

Therefore, students at high school need assistance to match their interests with the available universities and field of study programs. Moreover, students need to filter, prioritize, and efficiently get adequate information to overcome the web information overload issues.

The purpose of this study is to propose a novel hybrid system approach that guide high school students toward higher education and career choices that match their interests and preferences.

Recommender systems (RSs) are software that provide recommendations to active users [27]. These systems address information overload and help users to take decisions suitable to their interests. For example, when using a RS, users will have the option to select which product to buy, which movie to watch, or which article to read. Such software is commonly used when the volume of data outperforms the user's capability to analyze it.

Existing RSs can be essential tools in guiding high school students when searching for appropriate universities/colleges, and university majors that align with their aspiring careers. However, these systems have many limitations such as Cold-start, Sparsity, Gray-sheep, and Scalability problems [5]. For example, RSs that are based on the CF technique use users' ratings instead of supplementary knowledge of the products and users to generate recommendations [17]. Besides, in e-learning RSs, limitations occur when recommending specific choices of educational materials. One of the biggest limitations is that e-learning RSs have a weak capability to generate personalized recommendations. These limitations happen because of the variety in the studying style and education level of the learners [26].

Our contributions in this study are summarized as follows: Firstly, guiding high school students in making the right decision when selecting a university/college, university major, and career path. Secondly, proposing a novel KB hybrid RS approach supported by the CBR and ontology techniques. This hybridization strategy helps to generate recommendations based on users' knowledge and ratings. In our system, the CBR is integrated into the KB recommender engine to support the recommendation process and generate recommendations based on the prior university graduates' knowledge. Also, the ontology is integrated into the KB recommender system to represent the schools, higher education domain, career domain, and students' profiles models. Thirdly, enabling RSs to process high dimensional datasets that encompasses heterogeneous data types. Finally, overcoming and reducing the limitations of traditional RSs.

This study is organized as follows: an overview of the recommendation techniques, similarity metrics, neighborhood-based CF algorithms, evaluation metrics, and related works are discussed in section 2. The proposed hybrid RS approach and the study experi-

ments are presented in section 3. Finally, the conclusion and future work are discussed in section 4.

2. Background

In the following section, an overview of the recommendation techniques, similarity metrics, neighborhood-based CF algorithms, evaluation metrics and related works are presented.

2.1. Basic Recommendation Techniques

Many recommendation techniques were implemented to provide recommendations to users. Techniques such as Demographic-based [34], Knowledge-based [6] (Constraint-based [10], Case-based reasoning [15], Ontology-based [41]), Content-based filtering [37], Collaborative Filtering [20] (Memory-based [39,3], Model-based [31]), and hybrid RSs [8,16] are widely used in various domain. The following figure illustrates the taxonomy of the most popular RSs' techniques.

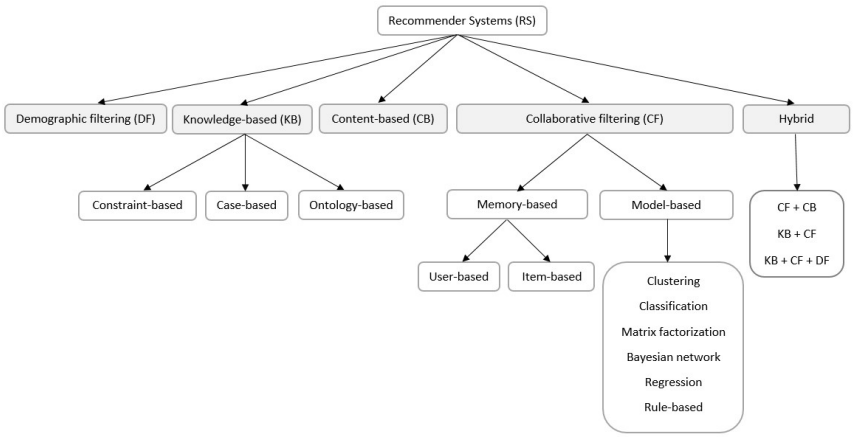


Fig. 1. Recommender systems' taxonomy

The CF recommender system is the most popular and successful approach implemented in the recommendation domain. This technique is based on the notion that if some users have the same preferences in their history, they will share mutual preferences in the future [12]. The CF technique integrates users' preferences, interests, and actions to suggest products to users based on the match between users' profiles. The CF algorithm encompasses two essential types namely the Model-based and Memory-based techniques. The memory-based computes the similarity between users based on users' activities, ratings, or selected items to generate appropriate recommendations. Memory-based integrates users and items' dataset to generate predictions. Besides, model-based calculates the similarities between users and/or items, then saves them as a model, and

then implements the saved similarity values to generate recommendations. The Model-based implements several algorithms such as clustering algorithm, matrix factorizations, Bayesian network or regressions

Nevertheless, the CF technique has many limitations and problems such as the Cold-start for new users, Scalability, Sparsity, Grey-sheep problems [8]. Additionally, CF technique faces many limitations such as treating heterogeneous data types and high dimensional datasets.

- Cold-start: the CF technique needs past users' history such as users' activities and ratings to generate precise recommendations. Cold-start problems occur when the dataset does not include sufficient ratings and preferences.
- Scalability: an enormous community of users and products exists in several of the environments that the CF systems make a recommendation in. Hence, great computation power is necessary to compute recommendations.
- Sparsity: this problem occurs when the items and users' matrix table is widely sparse. In this case, the precision of the recommendations will decrease since past users could not rate all the available products in the system.
- Grey-sheep problem: This problem is caused by odd recommendations since the user may have other features that do not match with any other user or community of users [7]. An example of a grey-sheep issue is when a user neither agrees nor disagrees with any user or group of users.
- Treating heterogeneous data types limitation: basic recommender filtering techniques have no capability to treat heterogeneous data types. Here comes the role of the hybrid recommender systems that can handle and compute heterogeneous data.
- Treating high dimensional datasets limitation: basic recommender filtering techniques have no capability to deal with high dimensional datasets. High dimensional datasets encompasses high number of attributes. This limitation can be addressed by decreasing the number of attributes in the dataset or using a Hybrid RS that can handle large datasets.

The CB technique works with the data provided by the user. Users' data is collected either explicitly by rating or implicitly by clicking a hyperlink. The CB algorithm function is to find products with the same content to suggest to the active users. The CB recommendations are based on what the active user liked. This recommender system compares the user's items ratings with items he or she did not rate and then computes the similarities. Based on that, the recommender system recommends the appropriate items, which are similar to the rated ones [32].

The KB recommender system generates appropriate recommendations based on explicit and implicit knowledge about the users and items. This technique integrates knowledge such as users' characteristics, preferences, interests, or needs [6]. KB recommender systems deal with the cases in which ratings are not used for the recommendations. Therefore, ontology is essential in the KB system to overcome the cold-start issues. Ontology is a KB technique, which does not take into consideration users and items past information. KB techniques are good examples to hybrid them with different recommendation techniques such as CF and CB.

DF recommender system purpose is to cluster the users based on their personal features in order to suggest appropriate recommendations. The DF recommendations are

based on users' demographic data such as age, gender, education, occupation, address (city, country), etc. [18,19]. The importance of such system is that it overcome the new user issue of the CF technique since they do not require user ratings. Also, it is easy to preprocess the data since it does not require domain knowledge. In DF, it is easy to identify similar users since new user must register and enter his/her demographic data to the system. In DF, users having the same demographic characteristics may also have similar preferences or tastes.

Combining two or more recommender techniques for a specific domain is an approach named "Hybrid RS". The hybridization method is commonly used for improving recommendation accuracy and overcoming the traditional RSs' problems and limitations.

Burke [8] classified the hybrid RS techniques into seven hybridization strategies namely (weighted, switching, mixed, feature combination, feature augmentation, cascade, and meta-level). In Weighted, the values of two or more RS are collected to generate a single recommendation. In Switching, the system navigates between the hybrid recommendations systems taking into account the running case. In Mixed, the output of two or more recommendation techniques are generated simultaneously. For example, CF rank (3) + CB rank (2) Combined rank (5). In Feature combination, the features from different sources are integrated into a single RS technique. In Cascade, the running recommendation technique refines the output of a second recommender system. In Feature augmentation, the output of one recommendation technique is integrated as input attributes into a second recommender system. In Meta-level, he model learned by one recommender is integrated as input into another recommender system.

In our work, we implemented the Feature augmentation strategy that combines the CF and KB techniques in a uniform system.

2.2. The Similarity Metrics, Neighborhood-based CF Algorithms, and Evaluation Metrics

Usually, people count on recommendations given by other people that are linked to different domains or products. Thus, RSs offer users the capability to count on the preferences or interests of large communities. In order to generate personalized recommendations, a RS makes some similarity evaluations on the users' preferences or interests and chooses which recommendations match users' tastes. So, what is the similarity between two items? In all situations, a full similarity is an absence of differences. Therefore, similarity metrics in a RS are about matching products or users that are most similar.

In this study, we aimed to implement and evaluate many RS algorithms and similarity metrics in order to generate better and accurate recommendations.

The Similarity Metrics: RS uses similarity metrics that are implemented in machine learning [14]. Most similarity metrics are correlated with vector space approaches. The applied Mahout Java library [1] has many similarity algorithms, which are,

- Euclidean Distance Similarity
- Pearson Correlation Coefficient Similarity
- Spearman Correlation Coefficient Similarity
- City Block Similarity
- Uncentered Cosine Similarity

The Euclidean Distance is the most common among all the distance measures. This distance is a straight-line distance between two vectors. The EuclideanDistanceSimilarity technique in mahout [1] java library calculates the similarity between two users X and Y. This technique considers items as dimensions and preferences as points along those dimensions. The distance is calculated using all items where both users have a similar preference for that item. It the square root of the sum of the squares of differences in position along each dimension. The similarity could be computed as 1 / (1 + distance) and the distance is mapped between (0, 1]. The distance between two points with coordinates (x, y) and (a, b) is given by

$$dist((x, y), (a, b)) = \sqrt{(x - a)^2 + (y - b)^2} \tag{1}$$

In Euclidean distance, the value of the distance is smaller when users are more similar. The larger the distance value is, the smaller the distance is. Thus, the closer the distance, the greater the similarity [28].

The PearsonCorrelationSimilarity is based on the Pearson correlation. The values for users A and B are calculated as follows:

- SumA2: the sum of the square of all A's preference values.
- SumB2: the sum of the square of all B's preference values.
- sumAB: the sum of the product of A and B's preference value for all items for which both A and B express a preference.

To calculate the correlation the following formula is used: sumAB / sqrt(sumA2 * sumB2).

$$sim(a, b) = \frac{\sum_p \in P(r_{a,p} - \overline{r_a})(r_{b,p} - \overline{r_b})}{\sqrt{\sum_p \in P(r_{a,p} - \overline{r_a})^2}\sqrt{\sum_p \in P(r_{b,p} - \overline{r_b})^2}} \tag{2}$$

a and b represents two users or items, p represents an item, $r_{a,p}$ and $r_{b,p}$ represent the user ratings from a and b for p, and average ratings of r_a and r_b are, for the item or user a and b [28]. Here the Pearson correlation coefficient is equal to the covariance of the two variables divided by the standard deviation of the two variables. The results range between [-1, 1], the larger the absolute value, the stronger the correlation, and the negative correlation has little significance for the recommendation.

The SpearmanCorrelationSimilarity is like the PearsonCorrelationSimilarity. However, the SpearmanCorrelation compares the relative ranking of preference values instead of preference values themselves. Each user's preferences are sorted and then assigned a rank as their preference value, with 1 being assigned to the least preferred item. The equation for Spearman Correlation Similarity is given in equation (3):

$$w(a, b) = \frac{\sum_{i=1}^{n}(rank_{a,i} - \overline{rank_a})(rank_{b,i} - \overline{rank_b})}{\sigma a * \sigma b} \tag{3}$$

The calculation in Spearman Correlation Similarity is very slow and there is a lot of sorting. Its results range between [-1.0, 1.0], 1.0 when there is a total match, -1.0 when there is no match.

The City block distance [13] also referred to as Manhattan distance. It calculates the distance between two points, a and b, with k dimensions. The City block distance is

computed like following:

$$\sum_{i=1}^{n} |a_i - b_i| \tag{4}$$

The City block distance result should be greater than or equal to 0. The result for identical points should be equal to 0 and greater than 0 for the points that express little similarity.

The UncenteredCosineSimilarity is an implementation of cosine similarity. Its result is the cosine of the angle formed between two vectors. The correlation between two points, a and b, with k dimensions is computed as:

$$Similarity = \frac{\sum_{i=1}^{n} a_i * b_i}{\sqrt{\sum_{i=1}^{n} a_i^2} * \sqrt{\sum_{i=1}^{n} b_i^2}} \tag{5}$$

This correlation ranges from (+1 to -1). The highest correlation is equal to +1 and the dissimilar points have a correlation equal to -1.

The Neighborhood-based CF Algorithms: The two types of neighborhood-based CF algorithms are the User-based CF and Item-based CF. The difference between the User-based CF and the Item-based CF is that User-based takes the rows of ratings matrix and Item-based takes the columns of ratings matrix for similarity measurement. In User-based, the item's recommendation rating for a user is calculated depending on those items' ratings by other similar users. The ratings are predicted using the ratings of neighboring users. In User-based, the Neighborhoods are defined by similarities among users. In Item-based, the item's rating is predicted based on how similar items have been rated by that user. The ratings are predicted using the user's own ratings on neighboring items. In Item-based, the Neighborhoods are defined by similarities among items.

The Evaluation Metrics: Many researchers found several evaluation metrics to evaluate the quality of the prediction. Prediction accuracy metrics find values that show how much the prediction is close to the real preference. The evaluation metrics help to assess the precision of the RS recommendations by comparing the predicted ratings with the rating of the active user. There are many prediction accuracy metrics used for testing the prediction accuracy of the used algorithms such as the Mean Absolute Error (MAE) [2] and Root Mean Squared Error (RMSE) [2]. In our graduates' dataset context, MAE and RMSE will assess how well the RS can predict a user's rating for a course/career.

The MAE metric evaluates the accuracy of an algorithm by comparing the value of predictions against the actual user's ratings for the user-item pairs in the test dataset. For each rating prediction pair, their absolute error is calculated. After summing up these pairs and dividing them by the total number of rating-prediction pairs, Mean Absolute Error can be found. It is the most commonly used and can be interpreted easily. The equation of Mean Absolute Error is given in equation (6).

$$MAE = \frac{\sum_{i}^{n} |r_i - e_i|}{n} \tag{6}$$

The RMSE is calculated by finding the square root of the average squared deviations of a user's estimated rating and actual rating. Once rating-prediction difference is calculated,

the power of 2 is taken. After summing them up and dividing them by the total number f rating-prediction pairs and taking square root of it, Root Mean Square Error can be found. The equation of Root Mean Square Error is given in equation (7).

$$RMSE = \sqrt{\frac{\sum_i^n (r_i - e_i)^2}{n}} \tag{7}$$

Where in both formulas for MAE and RMSE n is the total number of items, i is the current item, ri is the actual rating a user expressed for i, and ei is the RS's estimated rating a user has for i. The smaller RMSE and MAE are, the more accurate a RS. This is because RMSE and MAE will calculate smaller values if the deviations between actual and predicted ratings are smaller. By using evaluation metrics, prediction accuracy and efficiency of the CF methods can be calculated and compared.

The Evaluation Algorithms: To evaluate the accuracy of our recommender system, two evaluation algorithms were implemented namely the HoldOutEvaluator and SameSplitEvaluator.

- The HoldOutEvaluator method splits the case-base into two sets, one used for testing where each case is used as a query, and another that acts as a normal case-base. This process is performed several times.
- The SameSplitEvaluator method splits the case-base into two sets, one used for testing where each case is used as a query, and another that acts as a normal case-base. This method is different from the other evaluator because the split is stored in a file that can be used in following evaluations. This way, the same set is used as queries for each evaluation. The generateSplit() method does the initial random split and saves the query set in a file. Later, the HoldOutfromFile() method uses that file to load the queries set and perform the evaluation.

These two evaluation algorithms helped us to evaluate the accuracy of our proposed hybrid recommender system that is supported by the ontology and CBR system. In the following section, the related works are presented.

2.3. Related Work

Researchers have proposed several approaches for building RSs, which offer recommendations to users based on specific criteria that match their interests. For instance, Jihane Karim et al. [23] proposed a hybrid method for a generic and personalized CBR-based RS. The authors used a generic ontology to represent the essential knowledge needed throughout the reasoning task. Moreover, they proposed a hybridization technique that incorporates CBR and CF to enhance recommendations. The preliminary experiments for this system were performed using restaurant datasets. Additionally, several hybrid strategies with CF and DF techniques have been proposed in many studies [22,38,18]. This type of hybridization can minimize the limitations of the CF technique. Eventually, few hybridization approaches have been proposed with a combination of three filtering techniques such as CF, KB, and DF. For instance, the authors in this paper [21] proposed a hybridization strategy consisting of three core techniques namely the DF, semantic KB,

and CF. The goal of this RS is to enhance the visitor's experience in visiting museums and tourist places. The demographic approach is first used to overcome the CF cold-start issue and the semantic approach is then activated to provide recommendations to the user semantically close to those he/she has previously appreciated. Finally, the collaborative approach is used to recommend to active user works previously liked by similar users. In addition, several hybridization models have been implemented with the combination of the CF and KB to improve the accuracy of recommendations. The possibility of combining CF and KB techniques is introduced in [9]. This hybridization approach has the ability to overcome CF limitations such as the problem of new users or items. As an example, Chavarriaga et al. [11] proposed a CF and KB technique to suggest learning resources or activities. This approach helps learners to reach a high competency ranking by using an online course platform. Moreover, Jhon K. Tarus et al. [25] proposed a KB hybrid RS supported by sequential pattern mining (SPM) and ontology. This hybrid system provides recommendations of e-learning resources to learners. In this system, the ontology is integrated to represent the domain knowledge about the learner and learning resources. The role of the SPM algorithm is to determine the learners' sequential learning patterns. As well, Mohammad I. [33] designed and implemented a hybrid RS named OPCR that incorporates the CB and CF filtering supported by an ontology to overcome the user Cold-start problem. This system incorporates all information about courses and helping students to choose courses towards their career goals. Besides, Hsu Mei-Hua [30] presented an online-personalized English learning RS. This system is capable of suggesting students with reading lessons that fit their interests. This hybrid RS incorporates the CB, CF, and data mining techniques to study students' reading data and computes recommender scores. Besides, Rodriguez et al. [36] presented a student-centered Learning Object (LO) RS based on a hybridization approach that incorporates the CB, CF, and KB techniques. The LOs that are adapted to the learner model/profile are retrieved from the LO databases by implementing the saved descriptive metadata of the objects. Also, Tarus et al. [26] proposed an approach that combines the CF and KB supported by ontology to suggest personalized learning materials to online learners. In this system, the ontology is used to represent the learner characteristics while CF predicts ratings and provide recommendations. Furthermore, researchers have proposed several advanced approaches for building RSs based on cognitive models, sentiment, and affective analysis. For instance, this study [40] demonstrates that the analysis models of human cognition grips promise for the design of recommender mechanisms. Besides, Ha et al. [19] designed a multi-level sentiment network visualization mechanism based on emotional words in the movie domain. The proposed approach has been integrated into a RS to recommend movies with similar emotions to the watched ones. As well, the authors in this paper [24] studied sentiment and affective analysis in a RS of blogs. The blog's recommendations are based on the association between the sentiment and affective analysis of the blog and text content submitted by the users. In addition, this paper [29] presented an educational RS based on affective computing. The main aim of this RS is to discover educational resources based on emotion detection. Our general study of the RS techniques can be summed up by the advantages and drawbacks of the hybridization approaches shown in the following table.

Hybridization model	Advantages	Drawbacks
KB + Memory-based CF	- Reduce the cold-start problem of CF - Reduce the sparsity issue of CF - The accuracy of the recommendations of this hybridization outperforms the memory-based CF predictions. - Fast replying when user's preferences are modified.	- It is not scalable for large datasets. - It needs knowledge engineering.
KB + Model-based CF	- The accuracy of the recommendations of this hybridization outperforms the model-based CF predictions. - Fast replying when user's preferences are modified. - It has a scalability feature.	- The hybridization of the memory-based CF with the KB provided better results than this system. - It needs knowledge engineering.
DF + Memory-based CF	- Reduce the cold-start problem. - The accuracy of the recommendations of this hybridization outperforms the memory-based CF predictions.	- It is hard to acquire demographic data. - It is not scalable for large datasets.
DF + Model-based CF	- The accuracy of the recommendations of this hybridization outperforms the model-based CF predictions. - It has a scalability feature.	- The hybridization of the memory-based CF with the DF provided better results than this system. - It is hard to acquire demographic data.

Table 1. Comparison of different hybridization techniques

In summary, our approach differs from the previously mentioned approaches in the sense that it integrates the user-based CF and KB techniques that are supported by the CBR and ontology. This approach is named CBR and ontology-based hybrid recommender system (COHRS). The CBR and ontology knowledge are integrated into this hybrid system to overcome the issues and limitations of traditional RS. We integrated the ontology engineering to model the knowledge acquired from different resources such students' demographic data, interests, schools, universities/colleges, university majors, and career domain.

By incorporating the CBR and ontology into COHRS the following CF issues and limitations have been addressed:

– Grey-sheep problem: this problem is caused by odd recommendations since the user may have other features that do not match with any other user or community of users

[7]. An example of a grey-sheep issue is when a user neither agrees nor disagrees with any user or group of users.
– Treating heterogeneous data types limitation: basic recommender filtering techniques have no capability to treat heterogeneous data types. Here comes the role of the hybrid recommender systems that can handle and compute heterogeneous data.
– Treating high dimensional datasets limitation: basic recommender filtering techniques have no capability to deal with high dimensional datasets. High dimensional datasets encompasses high number of attributes. This limitation can be addressed by decreasing the number of attributes in the dataset or using a Hybrid RS that can handle large datasets.

Moreover, our proposed system is specialized in the field of guiding high school students toward higher education choices. No study has been conducted previously that describes the higher education domain with a hybridization strategy that combines the KB, CF, ontology, and CBR techniques. The following two tables present a comparison between COHRS and other hybrid RSs.

Hybrid RS Name	**Supported By CBR**	**Supported by Ontology**	**Applied to high dimensional dataset**
COHRS	X	X	X
Chavarriaga et al. Hybrid RS [11]			
Mohammad I. [33] OPCR		X	
Jhon K. Tarus et al. Hybrid RS [25]		X	
Hsu Mei-Hua Hybrid RS [30]			
Rodriguez et al. Hybrid RS [36]			
Tarus et al. Hybrid RS [26]		X	
Jihane Karim et al. [23]	X	X	

Table 2. The Core Techniques Implemented by Each Hybrid RS

Table 2 describes the core techniques implemented by each hybrid RS. Whereas, table 3 illustrates the hybridization strategy, key feature, targeted users and targeted domain of the compared hybrid RSs.

Hybrid RS Name	**Hybridization Strategy**	**Used for**	**Targeted users**	**Targeted domain**
COHRS	KB + User-based CF	Guiding high school student toward higher education choices such as university and field of study.	High school students	Higher education
Chavarriaga et al. Hybrid RS [11]	KB + Item-based CF	Recommending activities and resources that help students in achieving competence levels throughout an online course.	Online learners	E-learning
Mohammad I. [33] OPCR	CB + Item-based CF	Recommending personalized courses that match student's personal needs.	University students	Higher education
Jhon K. Tarus et al. Hybrid RS [25]	KB + Item-based CF	Generating recommendations of e-learning resources to learners.	Online learners	E-learning
Hsu Mei-Hua Hybrid RS [30]	CB + Item-based CF	Recommending students with English reading lessons that fit their interests.	Online learners	E-learning
Rodriguez et al. Hybrid RS [36]	CB + CF + KB	Providing learners with appropriate recommendations adapted to their preferences and bringing LOs closer than expected.	Online learners	E-learning
Tarus et al. Hybrid RS [26]	CF + KB	Suggesting personalized learning materials to online learners.	Online learners	E-learning
Jihane Karim et al. [23]	KB + Item-based CF	Recommending personalized items to customers	Customer	Restaurants Domain

Table 3. The Hybridization Strategy, Key Feature, Targeted Users and Targeted Domain

In the following section, the architecture of the proposed hybrid RS, similarity metrics, neighborhood-based CF algorithms, evaluation metrics, and experimental procedure are presented.

3. The Proposed Approach

The proposed hybrid system incorporates four core techniques (KB, user-based CF, CBR and ontology). This hybrid RS focuses on recommending universities/colleges, university majors, and career choices.

3.1. The Implementation Phases

This hybrid approach involves four main implementation phases: (1) the data acquisition phase, (2) the data-preprocessing phase, (3) the ontology design phase, and (4) the hybrid recommendation phase.

Fig. 2. The Four Main Implementation Phases

The Data Acquisition Phase: Our study focuses on analyzing university graduates trajectories and finding solutions to assist high school students to take appropriate decisions toward higher education choices. However, this study requires special types of data to be used in the analysis phase. Unfortunately, the required data is not available anywhere, since it is related to university graduates educational trajectories. In addition, it is very difficult to acquire it online and users are reluctant to disclose it. Data can be acquired in two ways: implicitly or explicitly. Explicit data are acquired from user ratings; for example, after listening to a song or watching a movie, and the implicit data are acquired form purchase history, search engine searches, or users/items' knowledge. In our case, we worked on gathering the required data through the explicit method. Therefore, we disseminated an online survey that includes more than 50 questions. The survey purpose is to reach university graduates and collect information about their educational trajectories, interests, current career occupation, etc. The dissemination process of the survey covered the Lebanese university graduates.

Our survey was published online in a period of 6 months and it involved questions of heterogeneous data types such as nominal, ordinal, numerical and open-ended. Ordinal data take their values in an ordered finite set. For example, a survey may ask the user to provide feedback on the service he/she received in a restaurant. The quality of service is ranked as (1) Not at all Satisfied, (2) Partly Satisfied, (3) Satisfied, (4) More than Satisfied and (5) Very Satisfied. The larger the set of values, the more informative the data. Nominal data names somewhat without assigning it to an order in relation to other numbered

items of data. For example, ”acting”, ”camping” or “cycling” classification for each user’s hobbies. Numerical attributes with continuous values that are represented by numbers and have most of the characteristics of numbers. Open-ended questions are questions that ask an applicant to answer in their natural language. They require a longer response. Thus, open-ended questions provide more information than a simple yes or no answer.

Our survey collected a real-world dataset that includes about 1000 university graduate applications and approximately 20,000 high school course ratings. This real-world dataset has varied data such as demographic data, interests, education and career knowledge, and ratings. In our hybrid system the collected demographic data and domain knowledge are used in the KB system in order to overcome the limitations of traditional RSs while ratings are used in the CF system. A collection of question types was used in this survey such as multiple-choice, Likert scale, and open-ended questions. For example, the answers for “How would you rate your high school grades on the following (Biology, Chemistry, Physics, and Mathematics. . .)?” are Very Good, Good, and Poor/Not concerned. Similarly, answer options for “If you already changed your university major, why did you change it?” are badly advised, Lack of understanding, you were uninterested in courses, and you had new interests...

The survey was organized into six main sections namely the survey description, graduate personal information, graduate high school or vocational school information, graduate first attended university information, graduate interests and career information, and graduate current university major information. The following are some samples of questions copied from the survey sections:

- Graduate personal information section: What is your Gender? (Male or Female); Select the work of your father
- Graduate high school or vocational school information section: What high school did you attend? (High School or Private School); What high school subject did you like best?
- Graduate first attended university information section: What was your university major?; How effective was the teaching within your major at the university? (Very Effective, Somewhat Effective or Not So Effective)
- Graduate interests and career information section: What kind of job/career interests you?; Is your current job related to your university major?
- Graduate current university major information section: What degree are you currently pursuing? (Bachelor Degree, Master Degree, Doctoral Degree or Other); How many times (if any) did you change your major at university (current or before)?

The Data-Preprocessing Phase: Inappropriate and redundant information or unreliable and noisy data exist in all dataset. Thus, analyzing data that has not been carefully refined can generate inaccurate results. Here comes the role of data preprocessing that involves many important steps and techniques. Therefore, data preprocessing is an essential phase in the data mining process and machine learning projects. In our work, we applied the following data preprocessing techniques in order to clean and refine the acquired data from the online survey.

- Data Quality Evaluation: data must be checked for missing, inconsistent and duplicate values.

- Dataset Dimensionality Reduction: significant real-world datasets have a great number of attributes (features). Therefore, the dimensionality reduction technique's purpose is to reduce the number of features in order to make the processing of the data more tractable. Reducing the dimensionality of a dataset is done by defining new features which are an arrangement of the original features.
- Attribute Sampling: sampling is picking a subset of the dataset that we are studying. Analyzing the whole dataset can be too expensive considering the time and memory constraints. Implementing a sampling algorithm can aid in reducing the size of the dataset to a level where the analyst can use a better machine learning algorithm.

In order to initialize the data-preprocessing phase, we extracted the required data as a CSV file from the online survey. The following figure illustrates the process of filling, extracting, cleaning, refining and preparing the desired dataset using many data preprocessing methods and tools such as Weka, WordNet, and Levenshtein distance.

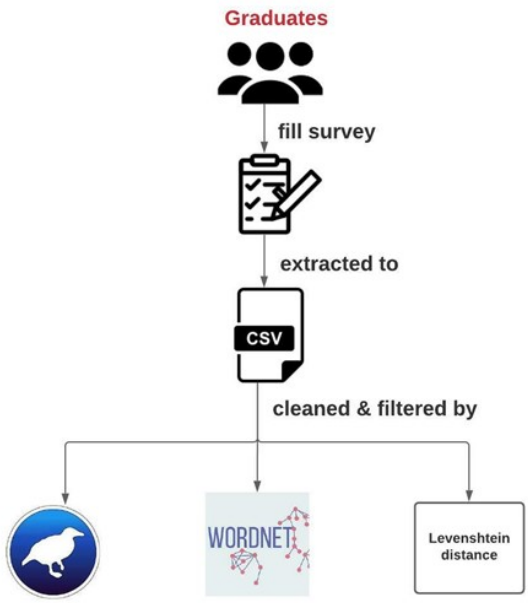


Fig. 3. Data pre-processing

As mentioned in the data acquisition phase, our survey contains more than 50 questions which led us to a large number of features. Therfore, we were obligated to reduce the dimensionality of our dataset. Thus, we used Weka InfoGainAttributeEval technique to perform feature selection by calculating the information gain for each feature for the output variable. The entry values range from 0 (means no information) to 1 (means maximum information). The features that give more information will get a higher information

gain value and can be chosen, whereas those that do not show much information will get a lower score and can be ignored in the analysis process.

Additionally, we implemented the WordNet that is a huge lexical database of many languages lexical. It is formed of sets of synonyms or synsets, which are groups of Nouns, Adjectives, Verbs, or Adverbs. The synonyms are linked based on lexical relationships, such as hyponym, hypernym, antonym, etc. This lexical database is available online for free download and usage. WordNet's structure enables this tool to deal with numerous tasks in NLP and computational linguistics such as information and document retrieval, improve search engine returns, automated document and text classification, Word sense disambiguation, machine translation, online lexical dictionary, etc. WordNet is usable from the R language to compute linguistic and text mining processing. As explained before, our survey contains open-ended questions. Thus, the graduates answered these questions in natural language and they expressed the same information differently. For instance, the term bike could be expressed as bike, bicycle, motorcycle, wheel, and cycle. In order to regroup our data, we used the WordNet database to find the synonym of the terms and strings that were entered by the graduates in their natural language. Getting the synonyms is required in the data prepressing phase in order to find the meaning of the terms and strings that describe the same object and then unify it in one common term. For example, the "IT manager" and "Information Technology manager" represent the same career domain. However, the clustering techniques will consider the "IT Manager" and "Information Technology Manager" as two different strings. Therefore, unifying the terms or strings into one common term can help the clustering technique to consider it as a same object and cluster it in the same group. A second example, the terms teacher and instructor share the same meaning but they are considered as two different terms in the clustering process. Nevertheless, when the synonym of the two terms is retrieved using WordNet and then unify it in one common term, the clustering technique will cluster it in the same cluster.

Moreover, many misspelled terms and strings were found in our survey entered by the university graduates. Therefore, we implemented the Levenshtein distance in order to compute the match between correct and incorrect terms and strings. The Levenshtein string metric was proposed by Vladimir Levenshtein. The Levenshtein distance measures the dissimilarity between two sequences. The distance between two strings is the least number of single-character alterations needed to transform one string into the other. This metric is applicable in sequence matching and spell checking. To clean our data and avoid loss information we used the Levenshtein distance. This method allows us to clean our dataset based on a reference dataset by computing the similarity between a source column from our dataset and a target column from the reference dataset that contains the correct terms and strings.

Furthermore, records of duplicate data should be deleted from the dataset before the analysis phase starts. Therefore, the Python dropduplicates function was applied to drop duplicate records from our survey data. Moreover, the multi-answer questions were split, using "," as a delimiter. Besides, we used the python Series.str.contains and Series.string.split functions to find specific terms and split each row in the series based a delimited.

The Ontology Design Phase: This phase represents the ontology that forms the model, individuals, and provides a semantic description of the education domain and career domain knowledge. The CBR recommender systems take advantage of this domain knowledge to obtain accurate results. The student and graduate's profile, school, higher education, and careers domains were described in an ontology using the Protégé OWL editor. This ontology is integrated into the KB recommender system in order to increase the accuracy of the recommendations. This ontology encompasses two main segments; the first segment is the Graduates that describes all the graduates' instances in the knowledge base such as their career interests, preferred courses, country, hobbies, etc. The second segment is the rest of the concepts in the ontology structure that describes all the attribute concepts of the student and education domain. Figure 4 illustrates the graph of our ontology. This figure represents the depth of the subclass hierarchy, which aids in the computation of the similarity measure.

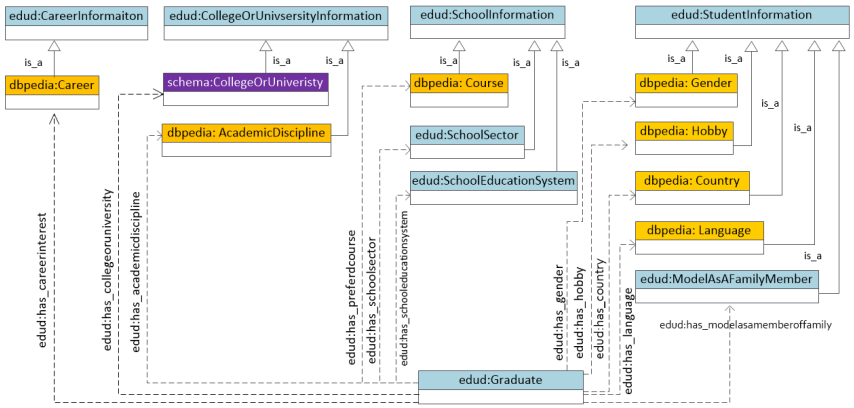


Fig. 4. The graph of the ontology design

The Hybrid Recommendation Engine Phase: Once the data and ontology are prepared, the RS computes the similarities and provides recommendations to the active student. The engine of this hybrid RS incorporates two core recommender systems namely the KB and CF illustrated in figure 6. The main function of the CF system is to compute the users' ratings and find similarities in order to generate appropriate recommendations. Then the output of the CF recommender system will be integrated as a new feature into the KB recommender system in order to recommend personalized recommendations. In the KB system, the semantic similarity is computed through the ontology structure based on the hierarchical order between the ontology concepts. This collaboration strategy between the KB and CF recommender systems is based on the "Feature Augmentation" hybrid strategy [7].

Researchers categorized the RS hybridization into two main cases:

– The first case is the uniform in which one RS algorithm has better precision than another algorithm over the entire space of recommendation. For instance, the Cascade

strategy with the stronger RS given higher priority, the Feature augmentation strategy in which the weaker RS algorithm performs as an assistant contributing a small amount of info, and the Meta-level strategy in which the stronger algorithm generates a heavy representation that reinforces the performance of the weaker algorithm.
– The second case is the non-uniform in which two recommender algorithms have different powers in different parts of the space. In this case, the process will need to be able to employ the two-recommender algorithms at different times. For instance, the Switching strategy is a natural choice here and needs the system should be able to detect when one algorithm should be favored. The Mixed and Feature combination strategies permit output from both RS algorithms without applying a switching measure.

The following figure illustrates the Feature augmentation hybrid procedure:

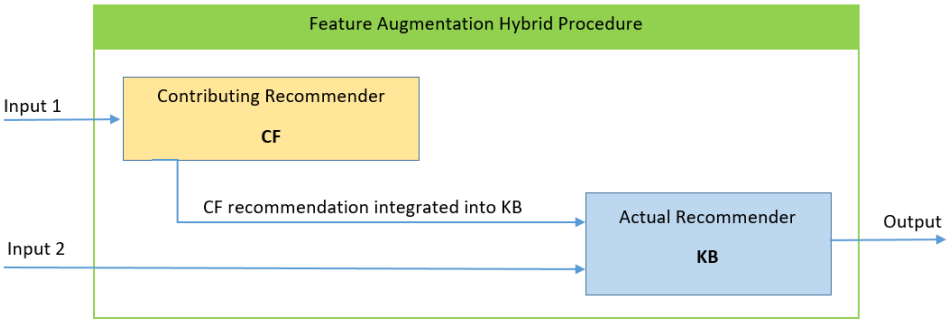


Fig. 5. Feature augmentation hybrid procedure

In our hybridization approach, we implemented the Feature augmentation technique because in our case the KB system is a stronger algorithm that is based on the domain knowledge and the CF is the weaker one that is based on the ratings. This approach enabled a contributing CF recommender to make a positive effect without interfering with the performance of the KB algorithm. Figure 6 shows the architecture of the proposed hybrid RS that integrates the User-based CF and KB algorithms. The proposed approach comprises 3 core modules described as follows:

– The first module illustrates the domain knowledge, which integrates the concepts and individuals of higher education, career, and students. The domain knowledge is formally represented in an ontology.
– The second module illustrates the hybrid RS engine, which incorporates the KB and User-based CF sub-systems. The CF system's role is to compute the k-most similar student and generate recommendations whereas the KB role is to generate the overall personalized recommendations to the active user based on ontology and CBR system.
– The third module illustrates the profile, rating, and query of the active student. In this module, the active user inputs his/her course/career's ratings and queries through a GUI in order to get recommendations. Course and career ratings are integrated into the CF system and the queries are integrated into the KB system.

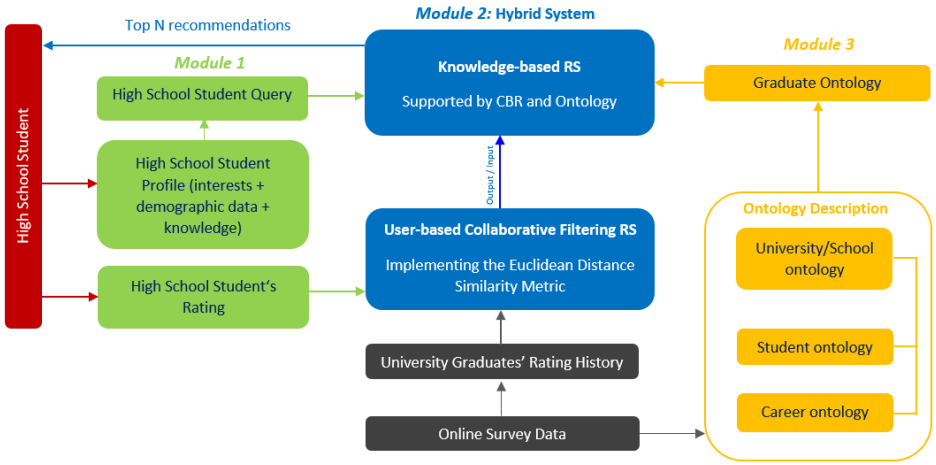


Fig. 6. The architecture of the proposed hybrid RS

These modules interconnect in a hybrid mechanism process and produce recommendations based on the active user's preference, interest, demographic data, and ratings. The following section presents the implementation and evaluation of the User-based CF and Item-based CF algorithms based on many similarity metrics, and the KB hybrid RS.

The Experimental Procedure A hybridization technique for basic RSs is needed to address the limitations and problems of some basic filtering approaches. Therefore, in this section, we implemented and evaluated the CF and KB algorithms in order to demonstrate the efficiency of our proposed KB hybrid RS. Thus, we experimented, evaluated, and tested the following three recommendation strategies:

- The User-based CF technique
- The Item-based CF technique
- The KB hybrid RS incorporated with the User-based CF technique and supported by the CBR and ontology.

The experimental study of the User-based and Item-based CF algorithms: In order to conduct the CF experiments, we extracted and refined a dataset from our online survey. This dataset encompasses 469 objects and 39 attributes. The objects represent the university graduates that have a job interest similar to their actual job. Besides, 39 attributes represent the items' ratings. This dataset contains about 11,000 ratings from 469 users on 39 items. All users in the dataset rated at least 20 items. The dataset involves correct real-world data that will ensure the accuracy of our experiments' returns. The correctness of our dataset is ensured by the way we disseminated and collected the survey entries. This survey was disseminated to university graduates that study in different disciplines and real employees that work in different domains. In addition, the data collection process involved face-to-face interviews to fill the intended survey. Since the selected dataset involve only graduates having a job interest similar to their actual job, we considered it a trusted real word dataset.

This experimental study divides the dataset into two sub-datasets. The first sub-dataset contains the training data and the second sub-dataset contains the testing data to test it. For each similarity metrics, evaluation has been implemented based on the MAE and RMSE. Since this experiment is based on item ratings, we implemented and evaluated the User-based and Item-based CF algorithms based on the Euclidean Distance Similarity, Pearson Correlation Similarity, Spearman Correlation Similarity, Uncentered Cosine Similarity, and City Block Similarity. The main function of the mentioned metrics is to find similarities between graduates based on their ratings. Then, the CF recommender system will recommend a career that is appropriate to the user's interests. In this experiment, some parameters have been determined such as the N neighborhood size, and training ratio of the experiment. In addition, the effects of different CF algorithms and similarity metrics were considered. The N neighborhood represents the nearest-neighbors to the object location. With user neighborhood, the RS can find the most similar user for the selected user. The training ratio represents the percentage of each user's preferences to use to produce recommendations; the rest are compared to estimated preference values to evaluate recommender performance.

To evaluate our CF recommender system, we implemented the mahout evaluation method [1]. The evaluate method evaluates the accuracy of RSs' recommendations. Applications will take a percentage of the preferences provided by the given DataModel as training data. This is classically most of the data, like 90 percent. This data is used to produce recommendations, and the rest of the data is compared against estimated preference values to see how much the recommender's predicted preferences match the user's real preferences. Precisely, for each user, this percentage of the user's ratings are used to produce recommendations, and for each user, the remaining preferences are compared against the user's real preferences. The return is a score representing how well the recommender's estimated preferences match real values. Lower scores mean a better match and 0 is a perfect match.

The experiment evaluation results based on different Neighborhood sizes: The size of the Neighbor can affect the prediction quality. By changing the number of neighbors, the sensitivity of the neighborhood is determined. In this section, the User-based and Item-based CF algorithms are evaluated and tested based on many similarity metrics and neighborhood sizes. The result of the experiments shows that the User-Based CF algorithm and the Euclidean Similarity metric have the lowest MAE equal to 0.45 and RMSE equal to 0.58, which means they predict better. All our experiments showed that the User-based CF algorithm and the Euclidean Similarity metric with Neighborhood size equal to 50 and Training Ratio equal to 0.8 have the lowest RMSE and MAE, which means they predict better. Therefore, we selected the Euclidean similarity metric as an appropriate technique for our CF recommender engine. The recommendations of the proposed User-based CF algorithm should be like:

- Recommended Academic Discipline: [Information Technology, value: 3.0]
- Recommended Academic Discipline: [Architecture and Construction, value:3.0]
- Recommended Academic Discipline: [Business, Management, and Administration, value: 3.0]
- Recommended Academic Discipline: [Finance, value: 3.0]

In the above recommendations, the term value represents the higher rate of the recommended item. The RMSE and MAE results are illustrated in figures 7 and 8.

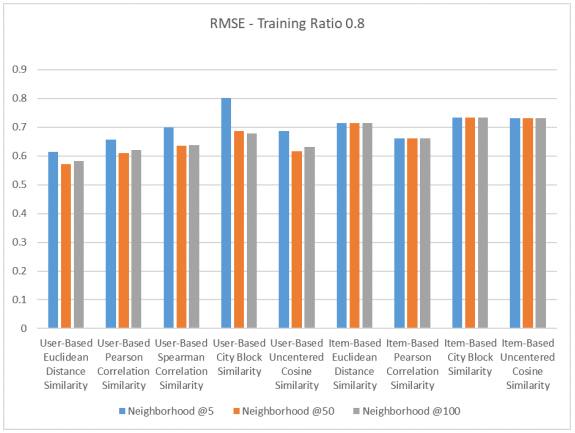


Fig. 7. RMSE for User-based and Item-based similarities with training ratio equal to 0.8

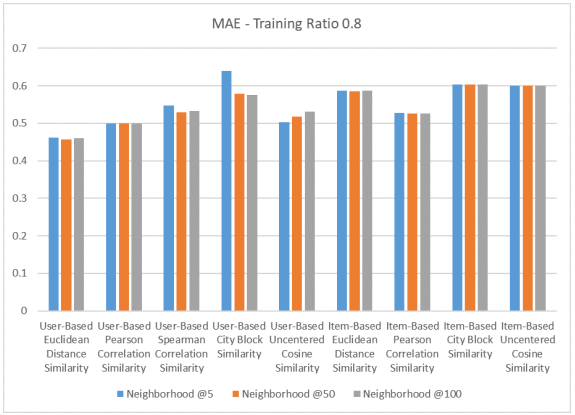


Fig. 8. MAE for User-based and Item-based similarities with training ratio equal to 0.8

At this point, the active student inputs his course ratings into the GUI of the system. This input permits the CF engine to generate recommendations based on the graduates' rating history. Then, the output of the CF system is integrated automatically as a new feature into the KB recommender system. This new feature will be used in the KB system as a support knowledge to the student interests' query. This query is submitted by the active student in order to feed the KB system. A query sample that describes a student's interests and demographic data is shown in Figure 9. Finally, the KB algorithm generates the top N recommendations based on the student's query.

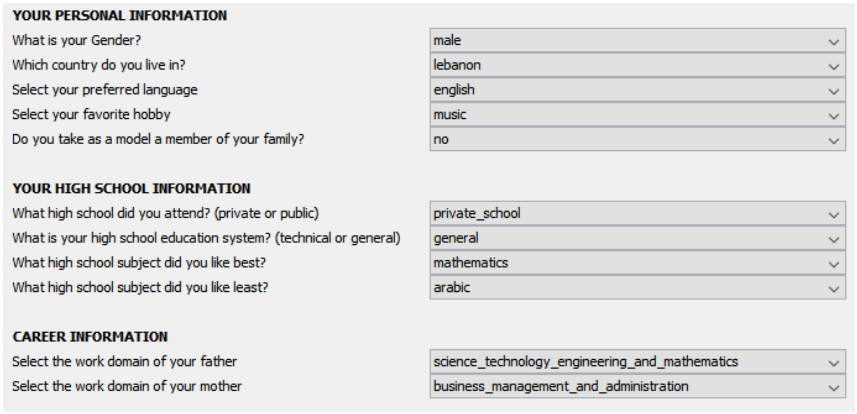


Fig. 9. Query sample

Figure 10 and 11 show that graduate case 584 and graduate case 601 are the most similar cases to the current active student query. If we compare the active student's query with the recommendations' result we notice that case 584 is more similar to the active student than case 601. This reveals that case 584 is totally similar to active student query and case 601 is diffirent in the hobby, mother work and graduate interest career attributes.

In our proposed hybrid system, the Feature augmentation strategy enabled the recommender engine to incorporates two separate types of recommender algorithms in a way that the output of the first recommender is fed into the input of the second recommender. In addition, the Feature augmentation strategy improved the performance of the proposed hybrid system and made a significant contribution to the quality of recommendations.

Finally, Hundreds of queries were processed by our hybrid RS approach. The experiments show the accuracy of COHRS based on two criteria namely the "accuracy of retrieving the most similar cases" and the "accuracy of generating personalized recommendations". To evaluate the accuracy of COHRS approach, the HoldOutEvaluator and SameSplitEvaluator algorithms were implemented. More information about these two algorithms are presented in section 2.2.4. The evaluation results show the high accuracy of COHRS based on using several percent of the dataset for testing and performing the process several times through many cycles. The evaluation results show the percentage of accuracy of COHRS based on many experiments as follows: 98 percent accuracy for

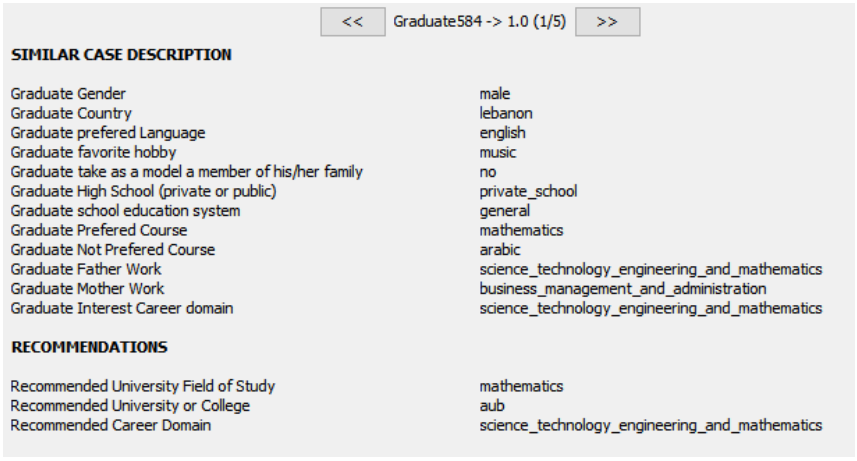


Fig. 10. Hybrid RS result (Top most similar case 584)

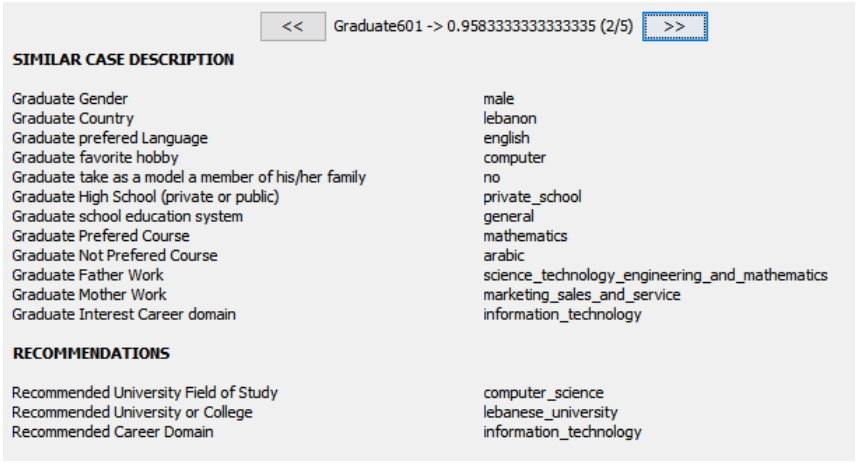


Fig. 11. Hybrid RS result (second most similar case 601)

"retrieving the most similar cases" and 95 percent accuracy for "generating personalized recommendations".

Besides, our analysis revealed that this hybridization approach is adequate to our high dimensional dataset that encompasses more than 50 heterogenious attributes. Furthermore, we noticed that the more knowledge we acquire the more effective the ontology-based hybrid RS could be. The novelty of our method focuses precisely on CBR and ontology-based hybrid RS within the higher education domain, of which to the best of our information, no research has been conducted using COHRS approach and presented this domain. Thus, we consider COHRS an effective approach for designing KB hybrid RS that support students in their higher education choices.

4. Conclusion

In this study, we proposed a novel hybrid RS approach named COHRS that incorporates the CF and KB recommendation techniques. This hybrid system is supported by CBR and ontology technologies. The purpose of this hybridization technique is to recommend to high school students appropriate universities/colleges, university majors, and career options. The recommendations of the proposed system are based on students' demographic data, course and career ratings, and the higher education domain. The experiments in this work enabled us to identify the appropriate similarity metrics, neighborhood size, and CF algorithm for our hybrid RS engine. Our contribution in this study is threefold: First, assisting high school students to find appropriate universities and university majors through a hybrid RS based on their interests and preferences. Second, proposing a novel hybridization approach that incorporates many recommendation algorithms. Third, minimizing the limitations of traditional RSs such as treating high dimensional datasets and heterogeneous data types. Additionally, solving problems that encounter most RSs such as the Gray-sheep problem. In future work, we will demonstrate the efficiency of the proposed hybrid RS approach by conducting more experiments and studying more RS algorithms and hybridization approaches.

References

1. Apache Mahout Essentials. Packt Publishing. ISBN:978-1-78355-499-7. (2015)
2. Badrul M. Sarwar, Joseph A. Konstan, Al Borchers, Jon Herlocker, Brad Miller, and John Riedl. 1998. Using filtering agents to improve prediction quality in the GroupLens research collaborative filtering system. In Proceedings of the ACM conference on Computer supported cooperative work - CSCW '98, 345–354. https://doi.org/10.1145/289444.289509. (1998)
3. Badrul Sarwar, George Karypis, Joseph Konstan,John Riedl.. Item-based Collaborative Filtering Recommendation Algorithms. (2001)
4. Basu, C., Hirsh, H. and Cohen W.. Recommendation as Classification: Using Social and Content-Based Information in Recommendation, in: Proc.the 15th National Conference on Artificial Intelligence, Madison WI 714-720. (1999)
5. Bobadilla, J., Ortega, F., Hernando, A., Bernal, J.. Acollaborative filtering approach to mitigate the new user cold start problem. Knowledge-Based Systems 26, 225-238. (2012)
6. Bruke, R.. Knowledge-based Recommender Systems. Encyclopedia of Library and Information Syst. 69(32). (2000)

7. Bruke, R.. Hyprid Recommender Systems. Survey and Experiments. User Modeling and User-Adapted Interaction 4:331-370. (2002)
8. Breese, J. S., Heckerman, D. and Kadie, C.. Empirical analysis of predictive algorithms for collaborative filtering. Computer Networks and ISDN Systems, pp. 43-52. (1998)
9. Burke, R.. Integrating knowledge-based and collaborative-filtering recommender systems. In Proceedings of the Workshop on AI and Electronic Commerce, pp. 69–72. (1999)
10. Burke, R.. Hybrid Web Recommender Systems. Springer Berlin Heidelberg, pp. 377 - 408. (2007)
11. Chavarriaga O., Florian-Gaviria B., Solarte O.. A Recommender System for Students Based on Social Knowledge and Assessment Data of Competences. In: Rensing C., de Freitas S., Ley T., Muñoz-Merino P.J. (eds) Open Learning and Teaching in Educational Communities. EC-TEL. Lecture Notes in Computer Science, vol 8719. Springer, Cham. (2014)
12. David Goldberg, David A. Nichols, Brian M. Oki, Douglas B. Terry.. Using collaborative filtering to weave an information TAPESTRY. Communications of the ACM 35(12):61-70. DOI: 10.1145/138859.138867.(1992)
13. Deeksha and S. Sahu, "Finding similarity in articles using various clustering techniques," 2017 6th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, pp. 344-347, doi: 10.1109/ICRITO.2017.8342449. (2017)
14. Desrosiers, C., Karypis, G.. A comprehensive survey of neighborhood-based recommendation methods, recommender systems handbook, (pp. 107–144). Berlin: Springer. (2011)
15. E. by David Leake. Case Based Reasoning. Experiences, Lessons and Future Directions. AAAI Press. MIT Press, USA. (1997)
16. F. S. Gohari, M. J. Tarokh.. Classification and Comparison of the Hybrid Collaborative Filtering Systems. Int. J. Res. Ind. Eng. 6 (2) (2017) 129-148. (2017)
17. G. Adomavicius, N. Manouselis, Y. Kwon.. Multi-criteria recommender systems in Recommender Systems Handbook, ed. by F. Ricci, L.Rokach,B. Shapira (eds.) (SpringerUS), pp. 769–803.(2011)
18. Gaurav Agarwal1, Himanshu Bahuguna2, Ajay Agarwal. Solving Cold-Start Problem in Recommender System Using User Demographic Attributes. International Journal on Emerging Technologies (Special Issue NCETST-2017) 8(1): 55-61. (2017)
19. Ha, H.; Han, H.; Mun, S.; Bae, S.; Lee, J.; Lee, K. . An Improved Study of Multilevel Semantic Network Visualization for Analyzing Sentiment Word of Movie Review Data. Appl. Sci. 2019, 9, 2419. (2019)
20. Herlocker, J.L., Konstan, J.A., Borchers, A. and Riedl, J.. An Algorithmic Framework for Performing Collaborative Filtering. Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 230-237. (1999)
21. Idir Benouaret. Un système de recommandation contextuel et composite pour la visite personnalisée de sites culturels. Autre [cs.OH]. Université de Technologie de Compiègne,. Français. ¡NNT : 2017COMP2332¿. (2017)
22. J.B. Schafer, D. Frankowski, J. Herlocker, S. Sen.. Collaborative filtering recommender systems. The Adaptive Web, 291-324. (2007)
23. Jihane Karim, Matthieu Manceny, Raja Chiky, Michel Manago, Marie-Aude Aufaure. Using Collaborative Filtering to Enhance Domain-Independent CBR Recommender's Personalization. IEEE 9th International Conference on Research Challenges in Information Science (RCIS), Athens, Greece. ⟨hal-02402483⟩. (2015)
24. João Pedro B. Ferreira, Franciscone L. A. Junior, Renata L. Rosa, and Demóstenes Z. Rodríguez. Evaluation of Sentiment and Affectivity Analysis in a Blog Recommendation System. In Proceedings of the XVI Brazilian Symposium on Human Factors in Computing Systems (IHC 2017). Association for Computing Machinery, New York, NY, USA, Article 25, 1–9. DOI:https://doi.org/10.1145/3160504.3160559. (2017)

25. John K. Tarus, Zhendong Niu, and Abdallah Yousif. A hybrid knowledge-based recommender system for e-learning based on ontology and sequential pattern mining. Future Generation Computer Systems 72: 37–48. https://doi.org/10.1016/j.future.2017.02.049. (2017)
26. John Tarus, Zhendong Niu, and Bakhti Khadidja. E-Learning Recommender System Based on Collaborative Filtering and Ontology. International Journal of Computer and Information Engineering 11, 2: 256–261. (2017)
27. Kantor, P. B., Rokach, L., Ricci, F., Shapira, B.. Recommender systems handbook. Springer, New York, Dordrecht, Heidelberg, London (Vol. I). ISBN 978-0-387-85819-7. DOI 10.1007/978-0-387-85820-3. (2011)
28. Keshav R, et al,. (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (3), 4782-4787. (2014)
29. LOPEZ, Maritza Bustos et al.. EmoRemSys: An educational recommender system by using emotions detection. RISTI [online]. n.17, pp.80-95. ISSN 1646-9895. http://dx.doi.org/10.17013/risti.17.80-95. (2016)
30. Mei-Hua Hsu. A personalized English learning recommender system for ESL students. Expert Systems with Applications 34, 1: 683–688. https://doi.org/10.1016/j.eswa.2006.10.004. (2008)
31. Minh-Phung Thi Do, Dung Van Nguyen, Loc Nguyen.. Model-based Approach for Collaborative Filtering. Ho Chi Minh city, Vietnam. (2010)
32. Mohammad Hamidi Esfahani, Farid Khosh Alhan. New Hybrid Recommendation System Based On C-Means Clustering Method". 5th conference of information Knowledge. IEEE. (2013)
33. Mohammed Essmat Ibrahim. An Ontology-based Hybrid Approach to Course Recommendation in Higher Education. 167. (2019)
34. Montaner, M., Lopez, B. and De la Rosa J.L.. A Taxonomyof Recommender Agents on the Internet, Artificial Intelligence Review, Kluwer Academic Publisher, 285 –330. (2003)
35. Owen S., Anil R., Dunning T. and Friedman E.: Mahout In Action. Manning Publications Co. ISBN 978-1-9351-8268-9. (2012)
36. Paula A. Rodríguez, Demetrio A. Ovalle, and Néstor D. Duque. A Student-Centered Hybrid Recommender System to Provide Relevant Learning Objects from Repositories. In Learning and Collaboration Technologies (Lecture Notes in Computer Science), 291–300. https://doi.org/10.1007/978-3-319-20609-7-28. (2015)
37. Pazzani M.J., Billsus D.. Content-Based Recommendation Systems. In: Brusilovsky P., Kobsa A., Nejdl W. (eds) The Adaptive Web. Lecture Notes in Computer Science, vol 4321. Springer, Berlin, Heidelberg. (2007)
38. Ruchika, Singh, Ajay Vikram, Sharma Dolly. Evaluation Criteria for Measuring the Performance of Recommender Systems, IEEE. (2015)
39. Sarik Ghazarian, Mohammad Ali Nematbakhsh.. Enhancing memory-based collaborative filtering for group recommender systems. DOI: 10.1016/j.eswa.2014.11.042. (2015)
40. Simone Kopeinik . Applying Cognitive Learner Models for Recommender Systems in Sparse Data Learning Environments. SIGIR Forum 51, 3 (December 2017), 165. DOI:https://doi.org/10.1145/3190580.3190608. (2018)
41. Z. Bahramiana, R. Ali Abbaspoura. An ontology-based tourism recommender system based on spreading activation model. Doi:10.5194/isprsarchives-XL-1-W5-83-2015. (2015)

Charbel Obeid is a Computer Science lecturer at AUL University - Lebanon. He received his Ph.D. degree in Computer Science from Claude Bernard University Lyon 1 - France. His research interests include artificial intelligence, data mining, recommender systems, Semantic Web, and ICT in education. He is the author of two journal papers and one conference paper. Also, he works as a technical engineer at 6SS a company that provides security solutions powered by Milestone XProtect.

Christine Lahoud is an Assistant Professor and the coordinator of the faculty of Engineering at the French University in Egypt, Cairo-Egypt. She obtained a Ph.D. degree in Computer Engineering from the University of Technology of Belfort-Montbeliard (2013), France. She joined then the department of Computer Engineering at the University of Reims in France and then Galatasaray University in Istanbul, as an assistant professor. She is currently doing research on recommendation systems. Her research interests include recommendation systems, knowledge management, semantic web, and artificial intelligence. She has been a member of the Program Committee of several conferences. She organized several workshops on Knowledge management as KARE 2011, EKM (2014, 2016, 2018), and a IJCEELL special issue on Knowledge Management Technologies in Education (2020).

Hicham El Khoury is an associate professor at the Lebanese University and a member of the LaRRIS Lab. He got his master's degree in Modeling from the Lebanese University and doctoral degree in Computer Science and Telecommunications from Paul Sabatier – Toulouse III. His research interests include Network Security Management Information, AI, clustering techniques, e-learning, EdTech, and ICT in Education. He is an author or co-author of 5 journal papers, 13 conference and workshop papers. He is also an ICT and Services Consultant at Lebanon-CERD (Center for Educational Research and Development) since 2017 and UK Lebanon Tech Hub since 2021.

Pierre-Antoine Champin is an associate professor at Université Claude Bernard Lyon 1, and a W3C fellow. His research interests include data and knowledge interoperability at web-scale, and the design of intelligent interactive systems. He has been representing Université de Lyon in various W3C working groups (Media annotation, RDF, LDP). He is an author or co-author of one book chapter, eleven journal papers, 71 conference and workshop papers, and four W3C standards. He is also a member of program committees in major Web and Semantic Web conferences (WWW, ISWC, ESWC).

Received: Jun 18, 2021; Accepted: February 27, 2022.

Computer Science and Information Systems 19(2):1007–1022 https://doi.org/10.2298/CSIS210609016P

A Machine Learning Approach for Learning Temporal Point Process

Andrija Petrović[1], Aleksa Biserčić[2], Boris Delibašić[2], and Dimitrije Milenković[2]

[1] Technical Faculty University Singidunum, Danijelova 32,
11000 Belgrade, Serbia
apetrovic@singidunum.ac.rs
[2] Faculty of Organizational Sciences, POB 52,
11000 Belgrade, Serbia
boris.delibasic@fon.bg.ac.rs

Abstract. Despite a vast application of temporal point processes in infectious disease diffusion forecasting, ecommerce, traffic prediction, preventive maintenance, etc, there is no significant development in improving the simulation and prediction of temporal point processes in real-world environments. With this problem at hand, we propose a novel methodology for learning temporal point processes based on one-dimensional numerical integration techniques. These techniques are used for linearising the negative maximum likelihood (neML) function and enabling backpropagation of the neML derivatives. Our approach is tested on two real-life datasets. Firstly, on high frequency point process data, (prediction of highway traffic) and secondly, on a very low frequency point processes dataset, (prediction of ski injuries in ski resorts). Four different point process baseline models were compared: second-order Polynomial inhomogeneous process, Hawkes process with exponential kernel, Gaussian process, and Poisson process. The results show the ability of the proposed methodology to generalize on different datasets and illustrate how different numerical integration techniques and mathematical models influence the quality of the obtained models. The presented methodology is not limited to these datasets and can be further used to optimize and predict other processes that are based on temporal point processes.

Keywords: temporal point process, Hawkes process, Poisson process, highway traffic prediction, ski injury prediction.

1. Introduction

Nowadays, one of the most popular research areas is focused on modelling event sequences. Event sequencing has become extremely popular in a wide range of applications such as road traffic estimation [1], epidemiology prediction [2], network activities [3], bioinformatics [4], e-commerce, etc. Event data carry information about event occurrence. Additionally, event data can also provide information about classes of events, types of events, participants, etc. This type of point process is known as a marked point process.

A point process is extremely useful in modelling traffic congestion and traffic event occurrences, e.g. arrival of vehicles, pedestrian movement, etc. [8]. Simulating highway traffic and predicting highway congestion is one of the main problems connected with point process modeling [9].

If compared with time series, event occurrences are treated as random variables generated in an asynchronous manner, which makes them fundamentally different from the time series where equal and fixed time intervals are assumed. This property makes them useful in a wide variety of applications where discretizing events to a fixed interval would result in poor prediction performances and high computational cost.

Generally, there are two types of point process models: temporal (univariate) point process and spatial-temporal (multivariate) point process. In the case of the univariate point process, the objective is to model temporally correlated event occurrences, whereas in spatial-temporal point process the event occurrences are correlated in space and time. Generally, multivariate point process is mostly used in the analysis of protein patterns [5] and financial market predictions [6]. The general formulation of the point processes makes them available to model event occurrences, both continuous or discontinuous (with jumps). Additionally, the point process can be further generalized by stochastic differential equations to stochastic point process.

The main idea behind different types of point process models is hidden in modelling a conditional intensity function (CIF). A CIF can be interpreted heuristically as the expected number of events that are going to occur in an infinitely small timestamp (dt). CIF can be modelled as a constant (homogeneous process) or as a function of time (inhomogeneous process). Learning an intensity function from a given dataset presents one of the most popular subjects of research [7].

In this paper, we present a data-driven approach for learning different types of CIFs used in temporal point process models. Our approach is based on the implementation of numerical integration methods for linearization of negative maximum likelihood (neML) in order to backpropagate derivatives of neML.

We tested our methodology on two real-life datasets that consisted of exact timestamps. The first dataset included highway toll passes recordings, and was a high-frequency dataset. The second dataset included timestamps when ski injuries occurred, and was a low-frequency dataset. Our methodology shows that it can be successfully used for various types of CIFs. Furthermore, four different baseline models based on neML scores: second-order Polynomial inhomogeneous process, Hawkes with exponential kernel, Gaussian process, and Poisson process were compared. The proposed methodology was evaluated on several metrics, amongst which is the minimization of negative log likelihood loss for demonstration of how well models fitted conditional intensity functions, Akaike information criteria (AIC), and the mean absolute error (MAE) for evaluating the quality of prediction for future time events.

To summarize, the contributions of this work are as follows:

- We presented a novel framework for learning different type of CIF in temporal point process based on implementation of one-dimensional numerical integration techniques for linearization of neML.
- The proposed method can be used with any kind of one-dimensional numerical integration technique.
- The method is tested on two real world datasets with high frequency and low frequency occurring events.

– The obtained results showed satisfactory performances on both datasets with respect to MAE, log likelihood and AIC.

The remainder of the paper is structured as follows. In section 2 the related work is reviewed. Background methodology, point process and Ogata's modified thinning algorithm are presented in section 3. A novel methodology for learning point process is presented in section 4. Experimental setup and results of real-world applications are presented in sections 5 and 6, respectively. The conclusions are drawn in section 7.

2. Related Work

We structure the discussion of the related work into two broad, previously mentioned, categories: intensity-based approaches and intensity-free approaches. The intensity-based approaches present methods where a point process is modelled by different functional forms of CIFs [10]. Intensity-free approaches present methods where a point process is modelled with some type of unsupervised machine learning algorithms.

Intensity-based approaches present the oldest approaches in point process modelling. They rely on a functional form that completely depends on the CIF. The Poisson process presents the simplest point process where conditional intensity function has a constant value [11]. The more complicated variant of this process is observed when the CIF is modelled as a product of kernels [12]. Recent research proposed different variants of modelling CIF by deep neural networks [7, 13]. Xiao et al. [13] presented an interesting approach of modelling CIF by a recurrent neural network. However, in this paper authors assume that integral in negative maximum likelihood is correlated only with the current timestamp. Even though this strong assumption cannot be justified by theoretical properties of point process models, the obtained results were significantly better compared to well-known baseline models. Chen et al. [14] and Zhang et al. [15] presented an interesting approach for modelling dynamics by deep neural networks. Moreover, the authors presented an interesting example where the point process was modelled by a differential equation and solved using the Euler method. Besides, the authors implemented the backpropagation technique for reducing memory complexity during the training phase.

Intensity-free approaches are based on modelling point processes by unsupervised learning techniques [16]. When compared to intensity-based approaches these methods can obtain better results, but they are more prone to overfitting due to smaller datasets or large expressive powers of the model. Variational autoencoders (VAE) present unsupervised machine learning algorithms that are mostly used for point process modelling. The Action Point Process variational autoencoder (APP-VAE) presents a variational auto-encoder that can capture the distribution over the times and categories of action sequences [17]. The APP-VAE obtained state-of-the-art results on the MultiTHUMOS and Breakfast datasets. A declustering based hidden variable model that leads to an efficient inference procedure via a variational autoencoder for solving multivariate highly correlated point process is presented by [18]. Besides VAE, generative adversarial networks (GANs) have recently been proposed as a method for describing event occurrences [19]. The authors proposed an intensity-free approach for point process modelling that transforms nuisance processes to a true underlying

distribution by using Wasserstein GANs. Experiments on various synthetic and real-world data substantiate the superiority of the proposed point process model over conventional ones. Compared to intensity-free approaches such as the GANs and VAEs, intensity approaches can provide information that is more explainable and interpretable in the case when CIF function is in linear form.

Applications of intensity-based point processes can be found in a wide variety of areas [35-37]. Point process are useful in medical care and health care. Liu et al. [28] presented an EM (expectation maximization) based point process for modelling of drug overdoses with heterogeneous and missing data. Additionally, a daily living activity prediction via combination of temporal point process and neural networks is presented in [29]. Besides medical care, point processes are also used in a wide variety of problems related to traffic [32-34]. A novel framework for modelling traffic congestion events over road networks based on spatio-temporal point process in combination with attention mechanism is presented in [30]. Motagi et al. [31] developed a self-exciting temporal point process to analyse crash events data and classify it into primary and secondary crashes. This model uses a self-exciting function to describe secondary crashes while primary crashes are modelled using a background rate function.

The model presented in this paper belongs to the class of intensity-based approaches. Compared with the standard intensity-based approaches, our model has more expressive power, whereas compared with intensity-free approaches it is less prone to overfitting. Moreover, our model can be easily applied to any type of conditional intensity functions, and linearization term can be also applied to very deep neural networks.

3. Framework

Based on the methodology proposed in this paper, we implemented a general framework for learning point processes. The framework consists of three distinct parts: Data cleaning and processing, model and hyperparameter selection, evaluation and simulation part. The framework is presented in Fig. 2.

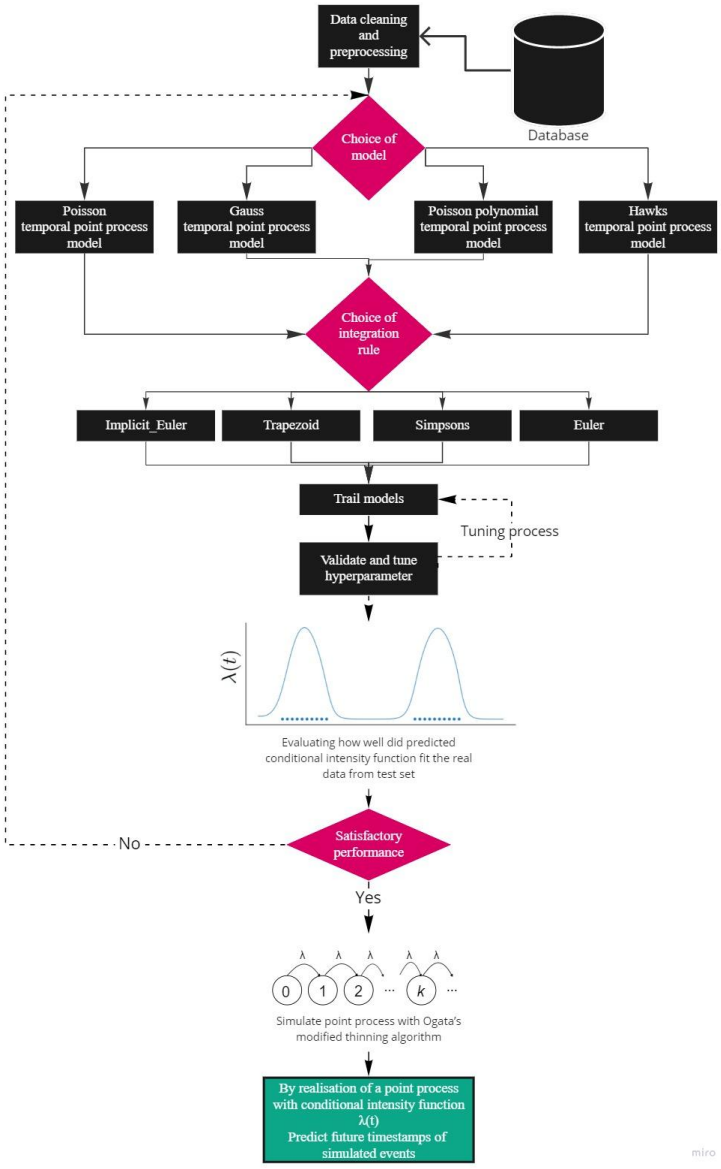


Fig 2. Framework for learning point processes

Data cleaning and preprocessing part consists of the methods that are used for cleaning and transforming raw data prior to processing and analysis. It is an important step that involves reformatting data, making corrections to data and making time sequences of occurred events. The transformation used in this part depends on problem formulations and raw data formats. Additionally, during this part the dataset is split on training, validation, and test set.

The first step in model selection and hyperparameter tuning step is to choose the point process model. In this framework the decision maker must choose between four different kinds of models: The Poisson temporal point process, the Gaussian point process, the Poisson polynomial process and the Hawkes process. The choice of a model primarily depends on the way the events are generated. Therefore, it is advisable to plot approximations of CIF with respect to moving windows and choose the model that best fits to it.

After model selection, the integration step and integration method must be chosen. In this framework, three different kinds of integration rules are presented: Implicit Euler, Trapezoidal, Simpsons and Gaussian quadrature method. Based on the integration rule, the integration step must be finely tuned in order to reduce the approximation error of the integral. Depending on the frequency of events, the integration step must be small and sufficiently large when event frequency is high and low, respectively. After model selection, the model is trained on training set.

The validation of integration step is applied with respect to log likelihood metric on the validation set. If the decision makers are satisfied with the obtained performances of the selected and validated models, they can proceed to the evaluation and simulation phase.

In evaluation and simulation part, the model is simulated and the results of simulations are used for testing the model on test set. Besides, log likelihood metric, mean absolute error is also used to evaluate model performances. Additionally, in combination with simulation, the obtained models can be further used in order to predict occurrence of the following events or to summarize some important statistical measures that can provide useful information to the decision maker.

4. Experiments

4.1. Experimental setup

In this section, we briefly present experimental setup, along with a detailed description of datasets and procedure for training and evaluation.

Datasets. The presented methodology was tested on two different datasets: a high frequency events dataset - traffic prediction on highway toll dataset, and on low frequency events dataset - prediction of ski injuries in ski resort Kopaonik. Therefore, in experiments we provided the methodology performance to learn event generation from two different types of datasets.

In the case of high frequency events dataset, the sequence of cars arriving at the ramp toll on the E 75 highway was taken as a concrete example of interest. Highway European Route E 75 is part of the International E-road network. The observed part connects two large Serbian cities - Belgrade and Niš. More precisely, the goal was to model the process of arrivals on the busiest ramp toll located at Niš from the Belgrade direction. The average time between two passes in one day is about 20 seconds, with a caveat that

the time between two passes is highly dependent on the time of the observed day. Standard 70/10/20 train, validation, and test splits were chosen respectively.

As for the low frequency point processes dataset, the observations of ski injuries in ski resort Kopaonik were taken as a concrete events of interest. Ski resort Kopanik is the biggest ski resort in Serbia. The dataset consists of records of ski injuries for the period from 2005 to 2020. Training and validation were done for period prior to 2020, and the test and evaluation were done for the year 2020.

Models. Four different well-known point process models were compared: Poisson temporal point process, Gaussian point process, Poisson, Polynomial point process, and Hawkes process. Additionally, each of these processes was combined with three distinct integration rules: Implicit Euler, Trapezoidal, and Simpsons.

Implementation: All **defined** models were implemented in Pytorch, an optimized tensor library for deep learning using GPUs and CPUs implemented in Python [22]. To run our experiments, we used a PC with the following configuration: Intel i9 CPU 9900K: 16 threads, 3.60GHz, 64 GB DDR4-2133, GPU RTX 3070 GPU 8GB GDDR6. Additionally, we provided the public repository with available implementation of the presented machine learning framework for learning point process.

Training. Due to heterogeneous nature of our benchmark datasets, both in the number of samples and **frequency** of the events, it was observed that we could get better results by fine-tuning the number of epochs (training time) and framework architecture (base model selection, integration rule, integration step) independently for both datasets. Validation dataset was used to choose hyperparameters (integration step, learning rate, etc.). Additionally, early stopping procedure evaluated on validation set was applied for obtaining best generalization performances of trained models.

Optimization. The Adam optimizer was used in order to fit the parameters of point processes. After hyperparameter tuning it was showed that all the models should be trained by 200 epochs, with a constant learning rate of 0.001, and integration step of 30. Each model was trained with the purpose to minimize negative log likelihood in order to reconstruct the true underlying event generation process.

Evaluation metrics. The models were evaluated on two key benchmark tasks. Firstly, we presented how well the models fitted real conditional intensity functions on the test set, or in the other words - how well the models performed minimization of negative log likelihood loss. Moreover, we evaluated Akaike information criteria (AIC) [27] to take in consideration model complexity. Secondly, using Ogata's modified thinning algorithm and conditional intensity function learned during training, we evaluated models prediction performances. The predicted time events were separated into bins of 3 different sizes: 5 minutes, 10 minutes, and 15 minutes for highway toll dataset, and the bins of 5 days, 10 days, and 15 days for ski injuries dataset, respectively. Then, for each binned period the mean absolute error (MAE) was calculated between the number of the predicted events and real number of events in that period. The visualization of sliding window approach for point process performance evaluation is presented in Fig. 3. In the case of high frequency dataset, due to dense event generation process, events were slid by 1 minute ($\sigma(t) = 1$ min), whereas in the case of low frequency dataset events were slid by 1 day period ($\sigma(t) = 1$ day).

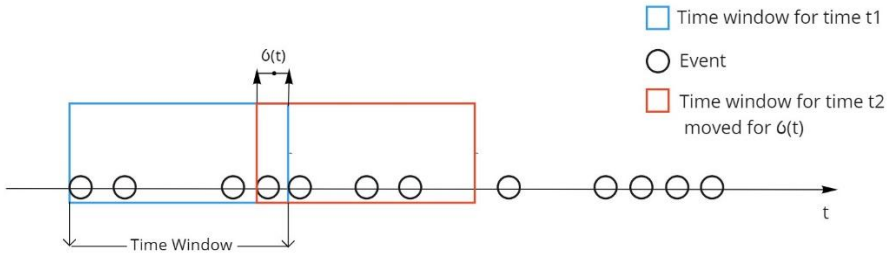


Fig. 3. Sliding window algorithm

Statistical tests. One way to prove that the obtained results are statistically significant is to apply two-sided "Welch" t-test [26]. "Welch" t-test is a two-sample location test used to test the hypothesis that two populations have equal means. Compared to standard Student's t-test, it is more reliable when two samples have unequal variances. One of the main conditions for applying t-test is samples independence assumption. Bearing in mind that the samples obtained by simulating point process are independent, first the time period is split in bins with fixed size. In each of these bins, the number of occurred events sampled from a point process model is counted and compared to the ground truth number of occurred events (absolute error (AE) is calculated). For each sample, the MAE error is calculated and samples obtained in this manner are completely independent from other samples, hence they can be used as inputs for two-sided "Welch" t-test. If the p-value in two-sided "Welch" t-test are less than 1% threshold, it can be stated that the means of the two groups (in this case MAE or two models) are unequal. If this is true, the results obtained by evaluation metrics are statistically significant.

4.2. Results and discussion

The prediction performances obtained by fitting different types of point process models with three distinct integration methods on highway car arrivals dataset are presented in Table 1.

Hawkes model with trapezoid numerical integration techniques, had the smallest log likelihood loss, AIC and the smallest MAE obtained in the case of all bin sizes. Furthermore, despite small training data and stochastic nature of data generation (i.e., dependency on part of day) it can be concluded that on average, the error of Hawkes model is less than half car per minute compared to the real car arrivals events. In addition, compared to the Hawkes process, Polynomial process obtained the worst results, whereas the results of Poission process are satisfactory, bearing in mind that the conditional intensity function is constant. Moreover, in Table 2, the results of two-sided "Welch" t-test are presented. In all presented models, trapezoid integration rule was used. It can be observed that p-values for each pair of models are less than 1% (0.01) threshold. Therefore, it can be concluded that the prediction means of each pair of models are unequal and results presented in Table 1 are statistically significant.

Table 1. Results of models performances with three distinct integration methods on highway car arrivals dataset

Bin_size	Model	Integration_	MAE	NLL (test_set)	AIC
5	Hawkes	Trapezoid	4.9	112.56	229.12
		Implicit_Euler	5.6	116.03	236.06
		Simpson	5.9	126.78	257.56
10		Trapezoid	8.8	112.56	229.12
		Implicit_Euler	10.3	116.03	236.06
		Simpson	9.7	126.78	257.56
15		Trapezoid	12.3	112.56	229.12
		Implicit_Euler	14.07	116.03	236.06
		Simpson	14	126.78	257.56
5	Gaussian PP	Trapezoid	6	178.85	363.7
		Implicit_Euler	5.9	216.44	438.88
		Simpson	5.7	156.25	318.5
10		Trapezoid	11.5	178.85	363.7
		Implicit_Euler	11.3	216.44	438.88
		Simpson	11.1	156.25	318.5
15		Trapezoid	17.2	178.85	363.7
		Implicit_Euler	16.6	216.44	438.88
		Simpson	15.4	156.25	318.5
5	Polynomial	Trapezoid	30.8	670.65	1347.3
		Implicit_Euler	63.4	756.04	1518.08
		Simpson	364	1206.34	2418.68
10		Trapezoid	61.7	670.65	1347.3
		Implicit_Euler	185.3	756.04	1518.08
		Simpson	729.3	1206.34	2418.68
15		Trapezoid	92.6	670.65	1347.3
		Implicit_Euler	336	756.04	1518.08
		Simpson	1094.3	1206.34	2418.68
5	Poisson	-	9.1	142.12	287
10	Poisson		17.7	142.12	287
15	Poisson		26.6	142.12	287

Table 2. Results of two-sided "Welch" t-test

Bin_size	Model 1	Model 2	t-statistic	p-value
5	Hawkes	Gaussian PP	-11.68	0.002
	Hawkes	Poisson	-78.19	0
	Gaussian PP	Poisson	-66.6	0.001
10	Hawkes	Gaussian PP	-45.04	0.001
	Hawkes	Poisson	-325.95	0
	Gaussian PP	Poisson	-291.207	0
15	Hawkes	Gaussian PP	-52.47	0.001
	Hawkes	Poisson	-347.21	0
	Gaussian PP	Poisson	-303.74	0

The results obtained by Hawkes model on highway car arrivals dataset are visualized in Fig 4. Firstly, it can be observed that the real and predicted conditional intensity functions are plotted with blue and green lines, respectively, by varying length of sliding window (Fig. 4a – 1 min, Fig. 4b – 5 min, Fig. 4c – 10 min). Additionally, the real and simulated timestamps of car arrivals were visualized as red dots. It can be concluded that despite being trained on just 70% of data, the model was pretty successful in predicting the real conditional intensity function. Moreover, the simulated car arrival events can imitate the real world application in the same manner.

In Table 3, the results of models performances on ski injuries dataset are presented. In the same manner, four different point process models performances with respect to three different numerical integration methods are showed. Gaussian point process model with Implicit Euler numerical integration techniques, had the smallest log likelihood loss, AIC, and MAE. The results of two-sided "Welch" t-test are presented in Table 4. The trapezoid integration rule was used in Hawkes model, whereas Implicit Euler rule was used in Gaussian point process. Based on the small p-values, it can be concluded that the prediction means of each pair of models are unequal and these results are presented in Table 1 and are statistically significant.

Again, the Polynomial process obtained the worst results. The Gaussian point process on average achieved MAE of 1.6 in the period of 5 days, which means that the Gaussian point process is going to predict on average 1.6 injuries more or less compared to the real number of injuries. Based on this, it can be emphasized that it is necessary to fit different point process models in order to find the one that best explains the true underlying distribution of event generation.

In addition, the results obtained by Gaussian point process on ski injuries dataset are visualized in Fig 5. Firstly, it can be observed that the real and predicted conditional intensity functions are plotted with blue and green lines, respectively, by varying length of sliding window (Fig. 5a – 5 days, Fig. 5b – 10 days, Fig. 5c – 15 days). Additionally, the real and simulated timestamps of ski injuries were visualized as red dots. It can be observed that real and predicted conditional intensity functions look very similar, and timestamps of the simulated events correspond to the timestamps of the real events.

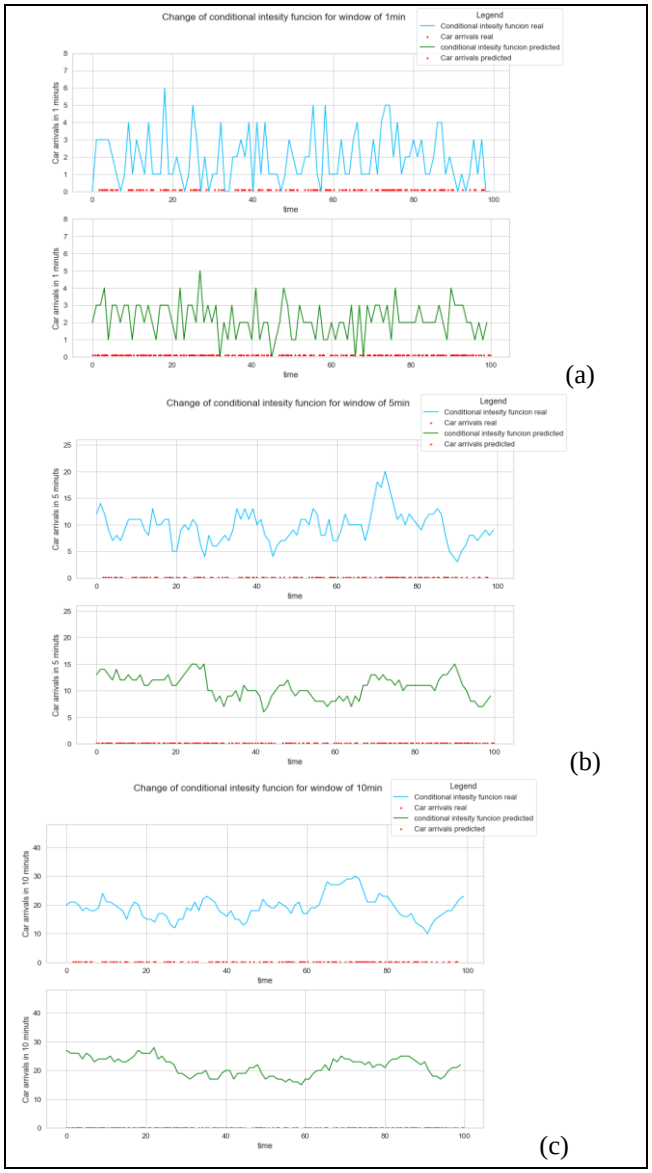


Fig 4. Visualization of how conditional intensity function predicted by Hawk's point process model fitted the real intensity function

Table 3. Experimental results - ski injuries dataset

Bin_size	Model	Integration	MAE	NLL (test_set)	AIC
5	Hawkes	Trapezoid	3.2	76.20	156.4
		Implicit_Euler	3.3	78.81	161.62
		Simpson	4	77.82	159.64
10		Trapezoid	4.8	76.20	156.4
		Implicit_Euler	6.2	78.81	161.62
		Simpson	8	77.82	159.64
15		Trapezoid	7.6	76.20	156.4
		Implicit_Euler	10	78.81	161.62
		Simpson	10	77.82	159.64
5	Gaussian PP	Trapezoid	3.8	99.64	205.28
		Implicit_Euler	1.6	75.42	156.84
		Simpson	2.8	88.56	183.12
10		Trapezoid	5.8	99.64	205.28
		Implicit_Euler	3.8	75.42	156.84
		Simpson	5.4	88.56	183.12
15		Trapezoid	9.6	99.64	205.28
		Implicit_Euler	4.9	75.42	156.84
		Simpson	7.3	88.56	183.12
5	Polynomial	Trapezoid	5.1	68223.36	136452.72
		Implicit_Euler	4.6	22480.22	44966.44
		Simpson	46	4262.03	8530.06
10		Trapezoid	9.8	68223.36	136452.72
		Implicit_Euler	6.6	22480.22	44966.44
		Simpson	102.4	4262.03	8530.06
15		Trapezoid	14	68223.36	136452.72
		Implicit_Euler	8	22480.22	44966.44
		Simpson	134	4262.03	8530.06
5	Poisson	-	3.1	142.00	286
10	Poisson		6.4	142.00	286
15	Poisson		12.3	142.00	286

Table 4. Results of two-sided "Welch" t-test

Bin size	Model 1	Model 2	t-statistic	p-value
5	Hawkes	Gaussian PP	-18.98	0.003
	Hawkes	Poisson	-87.71	0.001
	Gaussian PP	Poisson	-103.91	0
10	Hawkes	Gaussian PP	-15.87	0.002
	Hawkes	Poisson	-70.51	0.001
	Gaussian PP	Poisson	-88.08	0.001
15	Hawkes	Gaussian PP	-39.28	0.002
	Hawkes	Poisson	-103.03	0
	Gaussian PP	Poisson	-153.47	0

5. Conclusion

In this paper, we propose a new machine learning approach methodology for learning temporal point process based on the implementation of one-dimensional numerical integration techniques. The likelihood function of the point process has an integral of the CIF given in the limits of data observation. Bearing in mind that the CIF can take any kind of mathematical form, in many cases this integral is analytically intractable. Due to this, in this paper, we present an approach to linearize this integral with standard numerical techniques and to backpropagate the derivative through it. The presented approach was successfully tested on real-life data. The main disadvantage of this approach lies in high computational cost that is connected with backpropagation of derivative through each integration step. Therefore, this approach should be used only in the cases when point processes with analytically tractable integrals cannot obtain satisfactory prediction performances.

Furthermore, the methodology was evaluated on four different well-known point process models. In addition, we presented that different numerical techniques for integration can be successfully implemented in this framework. Moreover, we successfully simulated the obtained CIFs and compared them with the observed intensity functions.

Further studies should address using deep neural networks (feed-forward and recurrent networks) as a CIF to better capture dependencies between event occurrences.

Acknowledgment. This work was supported in part by the ONR/ONR Global under Grant N62909-19-1-2008. The authors would like to express their gratitude to the company Saga NFG d.o.o. Belgrade, for supporting this research. The authors would also like to thank public enterprises Roads of Serbia and Ski resorts of Serbia for providing data for this research.

References

1. Ryu, B., Steven, B. L. (1998). Point process models for self-similar network traffic, with applications. Communications in statistics. Stochastic models, 14(3), 735-761.
2. Zahrieh, D. (2017). Bayesian point process modeling to quantify excess risk in spatial epidemiology: an analysis of stillbirths with a maternal contextual effect.
3. Liu, S., & Wu, W. (2017). Generalized Mahalanobis depth in point process and its application in neural coding. The Annals of Applied Statistics, 11(2), 992-1010.
4. Farajtabar, M., Wang, Y., Gomez-Rodriguez, M., Li, S., Zha, H., & Song, L. (2017). Coevolve: A joint point process model for information diffusion and network evolution. The Journal of Machine Learning Research, 18(1), 1305-1353.
5. Jacobsen, S., Grove, H., Nedenskov Jensen, K., Sørensen, H. A., Jessen, F., Hollung, K., Søndergaard, I. (2007). Multivariate analysis of 2-DE protein patterns–Practical approaches. Electrophoresis, 28(8), 1289-1299.
6. Bowsher, C. G. (2007). Modelling security market events in continuous time: Intensity based, multivariate point process models. Journal of Econometrics, 141(2), 876-912.
7. Mei, H., Eisner, J. M. (2017). The neural hawkes process: A neurally self-modulating multivariate point process. In Advances in Neural Information Processing Systems (pp. 6754-6764).
8. Jia, R., Jiang, P., Liu, L., Cui, L., Shi, Y. (2018). Data driven congestion trends prediction of urban transportation. IEEE Internet of Things Journal, 5(2), 581-591.
9. Nguyen, T. T., Krishnakumari, P., Calvert, S. C., Vu, H. L., & Van Lint, H. (2019). Feature extraction and clustering analysis of highway congestion. Transportation Research Part C: Emerging Technologies, 100, 238-258.
10. Rasmussen, J. G. (2011). Temporal point processes: the conditional intensity function. Lecture Notes, Jan.
11. Last, G., Penrose, M. (2017). Lectures on the Poisson process (Vol. 7). Cambridge University Press.
12. Kirchner, M. (2017). An estimation procedure for the Hawkes process. Quantitative Finance, 17(4), 571-595.
13. Xiao, S., Yan, J., Yang, X., Zha, H., & Chu, S. M. (2017, February). Modeling the intensity function of point process via recurrent neural networks. In Thirty-first aaai conference on artificial intelligence.
14. Chen, R. T., Rubanova, Y., Bettencourt, J., Duvenaud, D. K. (2018). Neural ordinary differential equations. In Advances in neural information processing systems (pp. 6571-6583).
15. Zhang, T., Yao, Z., Gholami, A., Gonzalez, J. E., Keutzer, K., Mahoney, M. W., Biros, G. (2019). ANODEV2: A Coupled Neural ODE Framework. In Advances in Neural Information Processing Systems (pp. 5151-5161).
16. Ghahramani, Z. (2003, February). Unsupervised learning. In Summer School on Machine Learning (pp. 72-112). Springer, Berlin, Heidelberg.
17. Mehrasa, N., Jyothi, A. A., Durand, T., He, J., Sigal, L., Mori, G. (2019). A variational auto-encoder model for stochastic point processes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 3165-3174).
18. Yuan, B., Wang, X., Ma, J., Zhou, C., Bertozzi, A. L., Yang, H. (2019, September). Variational Autoencoders for Highly Multivariate Spatial Point Processes Intensities. In International Conference on Learning Representations.
19. Xiao, S., Farajtabar, M., Ye, X., Yan, J., Song, L., Zha, H. (2017). Wasserstein learning of deep generative point process models. In Advances in Neural Information Processing Systems (pp. 3247-3257).
20. Hawkes, A. G. (1971). Spectra of some self-exciting and mutually exciting point processes. Biometrika, 58(1), 83-90.

21. Xu, H., Farajtabar, M., Zha, H. (2016, June). Learning granger causality for hawkes processes. In International conference on machine learning (pp. 1717-1726).
22. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... Desmaison, A. (2019). Pytorch: An imperative style, high-performance deep learning library. In Advances in neural information processing systems (pp. 8026-8037).
23. Palm, C. (1988). Intensity variations in telephone traffic. North-Holland.
24. Radunović, D, Numeričke metode (2004). Akademska misao, Beograd.
25. Euler, L. (1768). Institutionum calculi integralis, vol. 1. imp. Acad. imp. Saent.
26. West, R. M. (2021). Best practice in statistics: Use the Welch t-test when testing the difference between two groups. Annals of Clinical Biochemistry,
27. Portet, S. (2020). A primer on model selection using the Akaike Information Criterion. Infectious Disease Modelling, 5, 111-128.
28. Liu, X., Carter, J., Ray, B., Mohler, G. (2021). Point process modeling of drug overdoses with heterogeneous and missing data. The Annals of Applied Statistics, 15(1), 88-101.
29. Fortino, G., Guzzo, A., Ianni, M., Leotta, F., Mecella, M. (2021). Predicting activities of daily living via temporal point processes: Approaches and experimental results. Computers & Electrical Engineering, 107567.
30. Zhu, S., Ding, R., Zhang, M., Van Hentenryck, P., Xie, Y. (2021). Spatio-temporal point processes with attention for traffic congestion event modeling. IEEE Transactions on Intelligent Transportation Systems.
31. Motagi, Samarth, et al. Point-Process modeling of Secondary Crashes. 2021.
32. Saha, A., Ganguly, N., Chakraborty, S., & De, A. (2019, April). Learning network traffic dynamics using temporal point process. In IEEE INFOCOM 2019-IEEE Conference on Computer Communications (pp. 1927-1935). IEEE.
33. Ganguly, N., & Saha, A. (2021). Modeling Inter-process Dynamics in Competitive Temporal Point Processes. Journal of the Indian Institute of Science, 101(3), 455-484.
34. D'Angelo, N., Adelfio, G., & Jorge, M. (2021, April). Some properties and applications of local second-order characteristics for spatio-temporal point processes on networks. In GRASPA 2021.
35. Tang, X., Li, L. (2021). Multivariate temporal point process regression. Journal of the American Statistical Association, 1-16.
36. González, J. A., Rodríguez-Cortés, F. J., Cronie, O., Mateu, J. (2016). Spatio-temporal point process statistics: a review. Spatial Statistics, 18, 505-544.
37. Reinhart, A. (2018). A review of self-exciting spatio-temporal point processes and their applications. Statistical Science, 33(3), 299-318.

Andrija Petrović is an assistant professor at the Technical faculty, Singidunum University. He received his first PhD in Decision making at the Faculty of Organizational Sciences, University of Belgrade and second PhD in Process engineering Faculty of Mechanical engineering, University of Belgrade. His research interests lie in machine learning, fairness in AI, and applications of machine learning in Chemical engineering.

Aleksa Biserčić is a PhD student at the University of Belgrade, Faculty of Organizational Sciences, with focus on Data Science and Deep learning. He has received a bachelor's degree in mechanical engineering and master degree in industrial engineering from University of Belgrade. His current work involves working as Data Scientist with US based software company Jaggaer and part time internship in laboratory

at University of Belgrade. His interests are in time series forecasting, natural language processing and recommendation systems.

Boris Delibašić is full professor at the University of Belgrade - Faculty of Organizational Sciences, Republic of Serbia. His research interests lie in data science, machine learning, business intelligence, multicriteria decision analysis, and decision support systems.

Dimitrije Milenković has received the B.S. degree in information systems and technologies in 2019 and the M.S. degree in information systems in 2020, both from the University of Belgrade. He worked on projects related to building analytical data platforms and data-driven products at TX Group (2019-2021) and Netcast (2017-2018). His research interests are data engineering, machine learning, decision-support systems, integration and interoperability of enterprise services and applications. Starting from 2020, he has been involved in the research projects of the Center for Business Decision Making at the Faculty of Organizational Sciences. Starting from 2021, he is a Research Associate in the Process Engineering Group of the National Institute of Standards and Technology and actively contributing to the development of Score, an open-source project for data exchange standards life cycle management.

Received: September 06, 2021; Accepted: February 16, 2022.

Computer Science and Information Systems 19(2):1023–1046 https://doi.org/10.2298/CSIS211215012O

Dynamic Network Modelling with Similarity based Aggregation Algorithm ⋆

Günce Keziban Orman

Computer Engineering Department, Galatasaray University,
İstanbul, TURKEY
korman@gsu.edu.tr

Abstract. Proper modelling of complex systems allows hidden knowledge discovery that cannot be explored using traditional methods. One of the techniques for such modelling is dynamic networks. In this work, we aim to develop a methodology for extracting proper dynamic networks. We concentrate on two fundamentally interconnected problems: first, determining the appropriate window size for dynamic network snapshots; and second, obtaining a proper dynamic network model. For the former problem, we propose Jaccard similarity and its statistical significance based compression ratio, and for the latter, we propose an aggregation approach that extracts dynamic networks with snapshots of varying duration. The aggregation algorithm compresses the system information when there is repetition and takes snapshots when there is a significant structural change. The experiments are realised on four simple or complex data sets by comparing our proposal with baseline approaches. We used well-known Enron emails as simple set and Haggle Infocomm, MIT Reality Mining, and Sabanci Wi-Fi logs as complex data sets. These complex sets like Wi-Fi or Bluetooth connections which are known to be noisy, making analysis difficult show the proximity of system objects. The experimental results show that the proposed methodology can be used to find not only significant time points in simple Enron emails, but also circadian rhythms with their time intervals that reveal the life-cycle of connected areas from complex Wi-Fi logs or bluetooth connections. According to testing on four real-world data sets, both compression ratios and the aggregation process enable the extraction of dynamic networks with reduced noise, are easy to comprehend, and appropriately reflect the characteristics of the system.

Keywords: dynamic networks, data compression, proper time intervals, algorithm.

1. Introduction

Dynamic network modelling of complex systems enables us to uncover previously unknown and emerging properties of the studied field [19], [27], [3], [14], [1],[24]. This model can be defined in different ways, such as link streams, event-based sequential graphs, or time flow-based sequential graphs. In [5], the authors evaluated the effectiveness of the different models by a data compression technique they proposed. The results demonstrated that different model types can be appropriate for different data sets. In this work, we concentrate on dynamic networks under the form of historically ordered

⋆ This is an extended version of INISTA 2021 conference paper ”Aggregating Time Windows for Dynamic Network Extraction”.

graph sequences. Each member of this sequence is called a snapshot, and it represents the interaction of the system objects for a given time interval.

Usually, a fixed duration, window size w, is used for determining the time interval. The topology of each snapshot depends on w. Since the structure of the model will affect all further analysis and temporal dynamics, i.e. community detection [21,8], link prediction, attribute prediction, change point detection [13], epidemic spreads, possibilities of communication and cascade of influence [20], it is crucial to find an optimal w for a proper dynamic network extraction. Yannick et al. underline that the larger the w, the higher the information loss related to temporal dynamics [20]. Previous works define an optimal w as being small enough to cover the temporal dynamics of the system but large enough to eliminate noise in the final model [12,26,29]. In this work, we concentrate on two inter-related problems; first, finding optimal w and second, extracting the proper dynamic network.

In most of the studies done so far, these two problems have been addressed together. Traditional methods consist of iterative and experimental procedures [29,30,18,12,16,13,7]. First, a set of candidates w is determined according to the domain knowledge and expertise of the authors. Second, candidate dynamic networks are extracted for each value of w. Third, the quality of candidate networks is measured for choosing the most appropriate one. Mostly, a dynamic network is represented in the form of its features' time series for measuring its quality. These methods differentiate from each other by first *choosing representative features* and, second, *measuring the quality of the time series*.

The most commonly used features for time series generation are network topological properties such as diameter, average distance, etc. [29]. We come across different time series analysis techniques for their quality measurement, such as discrete Fourier transformation [11], ARIMA [30] or using statistical metrics as variance or standard deviation [29]. Among them, the TWIN approach can be accepted as the baseline of traditional approaches [29]. It relies on basic optimisation of both the noise and variation level of the feature time series generated for different w. However, TWIN suffers from using non-scale invariant topological properties, i.e. diameter, in time series generation. This can be misleading when comparing different dynamic networks extracted for different candidates. A detailed discussion of this issue is done in our previous work, [22].

Recent studies have focused on using a reliable similarity metric to assess the stability of the snapshot sequence [6,22,16]. Darst et al. extract the snapshots for the different time intervals [16]. Time intervals between snapshots are determined by Jaccard similarity optimisation of consecutive events. The similarity of events is measured using an increasing time interval. By linear optimisation of similarity, it is decided which w is the most proper. Snapshots in the final network can have different w. There are more snapshots when the system is very active and a few snapshots when the system events are calm. Chiappori et al. propose a Jaccard similarity-based stability metric and a fidelity score [6]. They show that the extracted dynamic networks are unstable if filtering is not applied independently of w. They propose a parametric filtering procedure for removing less prominent links that are assigned as noise. Although the Jaccard similarity, as a scale-invariant metric, is a more objective measure than topological properties in comparing different w, the methods used in determining the stability of systems or change

point detection in the previous two studies are open to criticism because they do not use a reference value for comparing Jaccard values.

We have previously proposed using Jaccard similarity based scale invariant similarity metrics instead of topological properties in the TWIN for finding proper time intervals and demonstrated its effectiveness in [22,31]. In these works, the statistical significance limit of Jaccard is used as a reference point when a comparison is needed. Hence, a more objective selection is made at the best w determination. We have also proposed a procedure for determining a different duration for each snapshot extraction [31]. This procedure is similar to the one proposed in [16]. However, instead of optimisation, it determines the duration of the time interval by comparing the obtained Jaccard values with its statistical significance limit. Most of the traditional approaches use fixed durations for all snapshots [21,8,7]. However, sometimes there can be hectic periods where system members have a lot of interactions among themselves, while other times they stay calm. By considering this fact, our previously proposed procedure extracts snapshots with different durations.

This work can be seen as an extension of our previous work [31]. In the previous work, we first introduced a dynamic network extraction procedure that results in aggregated networks, second, evaluated its performance on two well-known data sets, Enron and Haggle Infocomm, and third, compared the aggregated networks with fixed-duration networks by using the TWIN. In this paper, we divide the problem into two parts: first, determining the optimal w, and second, extracting the appropriate dynamic network.

We propose three new contributions to this current work. Our first contribution is to define a new dynamic network compression ratio. It is based on Jaccard similarity and its statistical significance. The proposed compression ratio is a dynamic network global level metric. It is capable of comparing any two dynamic networks. We use it for determining the best w. Our second contribution is evaluating the effectiveness of both the new compression ratio and the previously proposed aggregation procedure on four data sets, including two additional sets; (1) MIT Reality Mining and (2) Sabanci University Wi-Fi logs. Our final contribution is comparing our proposal compression ratio with not only the TWIN but also with a new additional baseline, which is the stability metric proposed in [6] and also with the average link similarity of the consecutive snapshots of the dynamic network. These metrics are also Jaccard based. That is why it allows us to evaluate the performance of our compression ratio more clearly. Moreover, in this work, we clearly explain the behaviour of previously proposed similarity metrics by using toy examples and also present the algorithm of a previously proposed dynamic network extraction procedure.

In the following sections, the readers will find first, preliminaries with the details of baselines, i.e. TWIN [29], stability[6] and average link similarity, and second, an explication of our new compression ratio proposal with its algorithmic procedure of dynamic network extractor, respectively. Then, in section 3, we explain the experimental set up by giving the details of the studied data sets. In section 4, we first show and interpret the results of experiments for finding the best w by comparing the proposed compression ratio with baselines and, secondly, interpreting the result of the aggregation algorithm by comparing it with constant window size. Finally, in section 5, we explain the essential conclusions of this work and its future aspects.

2. Method

A complex system that evolves over time $\mathcal{C}$ is a system of interacting objects in which each interaction occurs at a time point in the continuous time interval defined between the beginning and the ending time points $[t_1, t_\theta]$. A dynamic complex network $\mathcal{G} = \langle G_1, \ldots, G_\theta \rangle$ is a sequential graph representation of $\mathcal{C}$ for discovering its emerging and non-linear dynamics. It is a consecutive set of static networks ordered historically. Each G_i $(1 \leq i \leq \theta)$ member of this set is referred to as a snapshot, where i representing a sub-interval in $[t_1, t_\theta]$. $G_i = (V_i, L_i)$ is a static and plain network in which V_i defines the set of nodes and $L_i \subseteq V_i \times V_i$ is the set of links. As in reality $\mathcal{C}$ is defined in continuous interval, we should discretize $[t_1, t_\theta]$ for extracting $\mathcal{G}$ for fitting its definition. Usually, a constant window size, w is determined for such discretization. Thus, finding the best window size w and extracting the proper $\mathcal{G}$ for the best w arise as prominent problems in this task. In the following part, we concentrate on these two problems. First, we explain the baseline methods in preliminaries. Second, we present our proposal for selecting w. Finally, we explain a simple but effective algorithm for automatically deciding time intervals for the snapshots based on previously selected w.

2.1. Preliminaries

In this part, we concentrate on two previously proposed approaches for selecting the optimal w: TWIN by [29] and Stability by [6]. Many previous approaches for determining the optimal w in the literature [29,30,4] employ TWIN as a baseline, whereas Stability is a novel approach.

TWIN is based on the combination of two time series statistics: *noise* and *compression*. For a given w and the corresponding $\mathcal{G}$, $F_w = \langle F_1, \ldots, F_t \rangle$ is a uni-variate time series of a topological property of $\mathcal{G}$. The noise of F_w is measured by its variance. Equation 1 gives the commonly accepted definition of variance where $\mu(.)$ describes the mean value of a studied uni-variate time series. Variance explains how much the F_w changes over time. The larger the variance, the noisier the signal of the studied topological property.

$$\sigma^2 = \frac{1}{t} \sum_{i}^{t} [F_i - \mu(F_w)]^2 \tag{1}$$

The compression ratio of F_w is defined as the string compression by run-length encoding. Hence, in TWIN, F_w is interpreted as a string. Run-length encoding compresses the repeating parts of strings. More clearly, if a character appears several times consecutively, run-length encoding represents it only once with the appearance count. Assume u represents the length of the string representation of F_w. Because each value in u represents a single character, the repetitive values can be compressed as if they were characters in a string. The length of the compressed string representation of F_w is defined as c. Therefore, the compression ratio can be calculated with the formula given in Equation 2. If the consecutive snapshots have the same value for the studied topological property, the compression ratio becomes large. Sulo et al. indicate that σ^2 and compression ratio, s, have opposite behaviours [29]. TWIN analyses σ^2 and s as functions of w. It determines the best w as the one with the smallest difference between σ^2 and s.

$$s = \frac{u}{c} \tag{2}$$

TWIN has three major flaws. For starters, network topological properties are not scale-invariant. As a result, using them to compare different $\mathcal{G}$ retrieved for different w can be misleading. Second, TWIN compresses numerical data using string compression. Run-length encoding checks only if consecutive members of a time series are the same or not. Although it can be used with integers, it has no application with real numbers. And yet, the majority of topological properties, such as density, average degree, and so on, have real values. As a result, only a few topological properties, such as diameter or radius, can be used in TWIN. In addition, variance is not a scale-invariant statistic. It is a popular method for determining the amount of noise in a time series. TWIN, on the other hand, uses it to compare distinct time series. When dealing with tiny changes, using a non-scale invariant metric might be misleading. In this work, we measure the noise level of time series with a normalised standard deviation over the mean when we need to compare them. Another baseline that we consider is the Jaccard Similarity-based Stability Metric proposed by [6]. Jaccard Similarity estimates the amount of shared parts between two sets in its broadest sense. The usage of Jaccard Similarity on link sets of consecutive snapshots is very common for measuring the stability of extracted networks [6,22,16]. In our study, we refer to this special form of Jaccard as *Link Similarity*, J_{link} (see Equation 3). Besides, *Node Similarity*, J_{node} can also be defined (see Equation 4) for consecutive snapshots' node sets. Both of them count the number of recurring links or nodes between two consecutive snapshots as well as the number of links or nodes that are seen at least in one of the consecutive snapshots.

$$J_{link}(t_i, t_{i+1}) = \frac{\mid L_{t_i} \bigcap L_{t_{i+1}} \mid}{\mid L_{t_i} \bigcup L_{t_{i+1}} \mid} \tag{3}$$

$$J_{node}(t_i, t_{i+1}) = \frac{\mid V_{t_i} \bigcap V_{t_{i+1}} \mid}{\mid V_{t_i} \bigcup V_{t_{i+1}} \mid} \tag{4}$$

In particular, *Stability* is defined as the weighted average of the link similarity over all consecutive snapshots. Its formula is given in Equation 5. In fact, Chiappori et al. propose a framework of two metrics; stability and fidelity [6]. However, because their fidelity measures the amount of lost information in the original data due to the snapshot extraction, we do not consider this metric. For instance, if two system actors are connected two times in a given duration, this is represented by one single link in the related snapshot by using plain networks. It can be classified as information loss based on any comparison of raw data and model. Here, using weighted links or shortening the duration of representing two connections in separate snapshots can be proposed as a solution. But it might be possible that the original data is noisy or includes redundant information depending on the applications. This is another crucial aspect to investigate, but it is out of the scope of our work. In this work, we extract snapshots in the form of plain networks. We do not concentrate on the compatibility between the extracted snapshots and raw data, but only on the snapshot's quality. That is why we use stability as another baseline, but we are not interested in fidelity.

$$I = \frac{\sum_{t_i,t_{i+1}} J_{link}(t_i, t_{i+1}).min(|L_{t_i}|, |L_{t_{i+1}}|)}{\sum_{t_i,t_{i+1}} min(|L_{t_i}|, |L_{t_{i+1}}|)} \tag{5}$$

Stability can be seen as a global metric that explains if the extracted $\mathcal{G}$ has overall stability. Because the J_{link} of consecutive snapshots is weighted based on the smallest snapshots' link amount, stability is more regulated to compared snapshots' sizes than the simple average J_{link}. However, it does not consider the effect of the number of common elements in the averaging scores. As a result, it can still be misleading when comparing different w. We consider both stability and average link similarity as baselines in our work. TWIN and stability are both useful in determining the best w which is the one resulting in the highest stability.

2.2. Proposed Method

Link and Node Compression Ratio for Window Size Selection The effectiveness of the usage of network similarity-based graph comparison metrics instead of topological properties is demonstrated in [22,31]. We describe how to choose the right w using novel compression ratios based on similarity. In the literature, there exist various distinct network similarity measures [9]. Nevertheless, because adjacency matrices are used, the majority of these metrics are dedicated to quantifying the similarity of two networks of the same size. When the number of nodes in the compared snapshots differs, which is the most common case, the union network matrices can be quite sparse. Thus, they suffer from the sparsity of the matrix. That is why we concentrate on Jaccard Similarity based metrics. Two different versions of Jaccard Similarity for measuring network stability are explained in the previous part with the equations 3 and 4. We use these two versions for the rest.

The Jaccard Similarity metric is a scale invariant metric with a range of 0.0 to 1.0. Its value is equal to 0.0 if there are no similar elements between the compared sets. In other words, the two sets are completely different. If it is equal to 1.0, it means that the two sets have exactly the same elements. If the network's similarity scores are high and there is a long-term repeat, then the network has redundant snapshots. Non-repeating signals, on the other hand, show variation in the snapshots. However, which value of Jaccard similarity indicates that two sets are sufficiently similar?

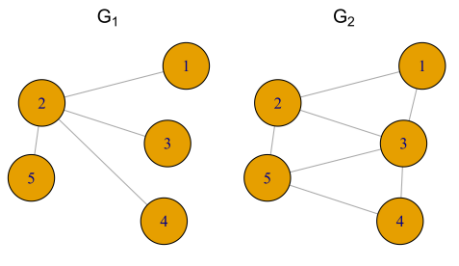


Fig. 1. Two consecutive simple snapshots

For example, the similarity scores for two consecutive snapshots in figure 1 are $J_{node} = 1.0$ and $J_{link} = 0.375$. The changes to the link structure have a direct impact on J_{link}. However, how can we tell whether these two consecutive snapshots are similar or not? We look up the statistically significant limit of Jaccard Similarity, which was previously investigated by [23]. They propose a probabilistic null-model based on the size of the compared sets. This model's score indicates the probability that two sets are similar by chance. In other words, if the Jaccard score of two sets is greater than a significant threshold, they are statistically similar. This null model (See Equation 6) assesses the randomness of two sets by taking into account both their common elements and the size of the smaller set.

$$P = \frac{\sum_{x=0}^{C} \binom{A+B-x}{x}}{\sum_{x=0}^{min(A,B)} \binom{A+B-x}{x}} \tag{6}$$

The numbers of elements in each set are defined as A and B, respectively, while the number of elements in common is defined as C. Because the maximum number of elements in common cannot be more than the minimum of the number of elements in A and B, P represents the probability of two consecutive sets having C elements in common by chance. In this study, A and B represent the number of node sets (or link sets) of the two network snapshots we compared. Similarly, C is the number of common elements in those sets.

If the compared snapshots are similar, their similarity must be greater than P, because being more similar by chance implies that the system's stability is still being maintained. As a result, this network segment can be compressed. The snapshots, on the other hand, are not similar if the similarity is smaller than P. We define D_w as the time series of the difference between the measured similarity and its statistical significance. Let $S_w = \langle S_1, \ldots, S_t \rangle$ denote a time series of a similarity metric, J_{link} or J_{node}, and $P_w = \langle P_1, \ldots, P_t \rangle$ denote a time series of statistical significance limits of each member in S_w. Then, the equation of D_w is given in Equation 7.

$$D_w = \langle S_1 - P_1, \ldots, S_t - P_t \rangle \tag{7}$$

Accordingly, we define a new compression metric s_{sim} as the ratio of t, which is the number of elements in uncompressed similarity time series S_w to t_{pos} the number of positive elements of D_w. Its equation is given in Equation 8.

$$s_{sim} = \frac{t}{t_{pos}} \tag{8}$$

These new statistics are called as *link* and *node compression* for the different similarity scores used in S_w. Because similarity scores are determined for consecutive snapshots, they do not explain the features of a snapshot but the stability of the system. That is why we do not consult the variance of similarity scores. We determine the best w as the one where the compression ratio is the lowest. That window size allows you to extract the most varied and not redundant snapshots. We show the effectiveness of the compression ratio by comparing their results with previously explained baselines in section 2.1.

Window Aggregation Algorithm for Proper Dynamic Network Extraction We previously proposed a window aggregation algorithm for extracting a proper dynamic network with a lower number of snapshots, having all the critical and necessary information for network analysis [31]. Here, we make small modifications to integrate the proposed compression ratios in the algorithm to increase its accuracy. The main idea behind this aggregation algorithm is that if a candidate snapshot is similar enough to already existing snapshots, we do not represent it separately in the final dynamic network but aggregate it into the existing model. The aggregation process is iterative and simple.

First, the best window size, w, is computed for the entire network by using the proposed compression ratio in the previous section, we generate the first snapshot, G_1 from the first time point of aggregation with a duration w. Then, for each following snapshot, G_i, we aggregate it and continue the iteration if G_i is similar to G_1. However, if it is not similar, we end the current aggregation process and start a new aggregation process for the rest of the time slices. The pseudo code is given in the algorithm 1.

Algorithm 1: *Window Aggregation Algorithm for Dynamic Networks*

Require: $\mathcal{C}, w, t_1, t_\theta$
Ensure: $\mathcal{G}$
1: $G_{init} \leftarrow extract(\mathcal{C}, t_1, w)$
2: $G_{aggr} \leftarrow G_{init}$
3: $t_i \leftarrow t_1$
4: **while** $t_i < t_\theta$ **do**
5: $G_i \leftarrow extract(\mathcal{C}, t_i + w, w)$
6: $sim \leftarrow S(G_{init}, G_i)$
7: $limit \leftarrow P(G_{init}, G_i)$
8: **if** $sim < limit$ **then**
9: $\mathcal{G} \leftarrow add(\mathcal{G}, G_{aggr})$
10: $G_{init} \leftarrow G_i$
11: **else**
12: $G_{aggr} \leftarrow aggregate(\mathcal{C}, G_{aggr}, G_i)$
13: **end if**
14: $t_i \leftarrow t_i + w$
15: **end while**

This process requires a complex system $\mathcal{C}$ that evolves over time, a time window size w and the beginning and ending time points t_1 and t_θ respectively. It returns a proper dynamic complex network; $\mathcal{G}$. Lines #1- 3 are devoted to the initialisation of the following objects; the inital snapshot, G_{init}, the base snapshot from which the aggregation will take place, G_{aggr}, and the time flow counter, t_i. At line #5, the snapshots are extracted sequentially in a loop for each coming time duration of length w into the variable G_i. After that, similarity and limit values are calculated at line #6, 7, by $S(.)$ and $P(.)$ respectively. One can use either J_{link} or J_{node} which are introduced in Equations 3 and 4 respectively, or their mean value for similarity. In our experiments, we adopt mainly to J_{link} for quantifying structural changes. For $P(.)$, we use the statistical significance given in Equation 6. The algorithm checks whether these two snapshots are similar enough to be

aggregated, i.e. their similarity score being larger than their statistical significance value at line #8. If they are not similar, the snapshots aggregated so far are added to $\mathcal{G}$ at line #9. Then, in order to continue the process, the current snapshot becomes the initial snapshot for the rest of the calculation from this point on (line #10). Otherwise, if the snapshots are similar, then they are aggregated to form a single snapshot that represents both of them at line #12. Finally, the time slice is shifted by the window size at line #14, in order to continue computation the next iteration till the end of the time interval.

Note that if w is too large, some critical time points where the system exhibits important change can be ignored. But, if w is too small, consecutive snapshots can be too similar, especially if the system is changing slowly. Here, we use the link similarity and its statistical significance, which are introduced in Equations (3) and (6) respectively. If the system is changing slowly, comparing consecutive snapshots can be misleading because the changes occur slowly and they cannot be captured in a short period of time. In order to capture these smooth and latent changes, we propose to measure the similarity of the current snapshot with the first snapshot that the aggregation process started with, rather than comparing consecutive ones. One of the most important points for the aggregation process is to decide if the measured similarity is critical. If the J_{link} score of the two snapshots under consideration, G_i and G_1, is greater than P, we aggregate G_i into the networks that have already been aggregated with G_1. An aggregated dynamic network can be seen as a compressed network including informative snapshots and not having redundancies. Although we use link similarity here, it can be applied with applied with any other network similarity metric having a significance threshold.

3. Experiments

We conducted experiments on four data sets: Enron Email, Haggle Infocomm, MIT Reality Mining, and Sabanci University Wi-Fi log data sets. These data sets are used in the studies dedicated to the same problem as us [7,29,22]. The statistics of these sets are given in Table 1.

Table 1. Data sets

Data Set Name	**Size**	**Time Span**
Enron Email [29,13,16]	151 employees	53 months
MIT Reality Mining [29,30,4,7,11,13,16]	90 users	9 months
Haggle Infocomm conference [4,13]	41 participants	4 days
Sabanci University Wi-Fi logs [22]	5378 devices	9 days

The Enron data set is an email set sent to or received by Enron employees between 1997 and 2003. We used the ”From” and ”To” fields of each email as nodes to generate snapshots with the timestamp. When there are multiple emails in the To field, we split them as if they were unique emails. Therefore, the clean data set consists of a date field, a single email address for the From field, and lastly, a single address for the To field. Each

of the From and To fields represents a node in the snapshots. Thus, every transaction is a link between these nodes.

The MIT Reality Mining data set consists of Bluetooth discovery data suggesting social interaction between people [10]. The data set was generated at the MIT Media Laboratory over a nine-month period. During the study, 93 participants were given Nokia 6600 cell phones, and a comprehensive data set was composed based on various features such as Bluetooth device discovery scans at five-minute intervals, voice and text messages, active applications, etc. We use only Bluetooth discovery data in which the source mac address and target mac addresses are captured with the timestamp. In our study, we represent each Bluetooth connection as a unique event. The source mac address and each target mac address are represented as nodes, and each transaction is a link between these nodes.

The Haggle Infocomm data set contains bluetooth discovery data like the MIT Reality Mining data set. It contains 98 unique participants, who have scanned 4724 unique devices during the Infocomm conference in 2006 in Barcelona. Special devices called iMotes are deployed throughout the area. Seventeen of them are long-range static iMotes; three of them have been placed in the lift of the hotel; the rest of the 98 are the participants of the workshop. Instead of the mac addresses, we use the IDs as nodes. Each row in the data set represents a discovery event. Hence, they are considered links between nodes. Since the experiment was done between April 24th and April 27th, 2006, and the iMotes were activated on April 23rd at 5:01 pm, we shifted the timestamp to match the correct dates.

Sabanci University Wi-Fi log data consists of system connection metadata on campus from the 2016–2017 fall semester, which was logged in every 10 minutes by Sabanci University's IT department. Every logging procedure consists of all devices' connection activity at the Wi-Fi access points in the campus. Thus, logging 10 minutes allows us to track the changes occurring in less than 10 minutes as well. However, the 10 minute period is critical because even if a connection continues longer than 10 minutes, it is represented as broken and reconnected in the data set. We fix this issue by preprocessing the data. Campus life shows a similar circadian rhythm for all periods. For the sake of comprehensibility, we only interpret nine representative days of the signals in this study, even if the data was collected for 137 total days. But the results and comments presented for the 9-day time period are valid and they can be expanded to the overall period of the data set. The records include the device ID, connection and disconnection timestamps, and the Wi-Fi Access Point (WAP) name. The device IDs were anonymized by assigning unique values to each MAC address connected to any WAPs on the campus. We create a node for each device ID that appears in the system. If two devices connect to the same access point for a given time interval, we put a link between them. A link between two nodes might be a sign that those devices are in the same place.

In [22], we have discovered circadian rhythms in the dynamic network representation of the daily cycle of Sabanci University campus life too. Similar circadian rhythms of human activity from electronic records were discovered by [2,25,15]. In [2], the authors underline two major periods of circadian rhythm; *day time* and *night*. Accordingly, there are cycling ascending and descending interaction activity. In our previous experiments, those periods corresponded to the hours of the day when the campus gets active and passive. Indeed, modelling these kinds of systems with a fixed w for both active and

passive periods results in the extraction of many redundant snapshots in the passive period. Also, there is a risk of missing important activities in the active period.

We consult to the previous studies for determining the candidate w values. Accordingly, for Enron, the duration less than 1 day results unrealistic snapshots having several unconnected components with sparse link structure. That is why we consider one day and seven, fifteen, thirty, sixty and hundred eighty days for w. For MIT Reality Mining, we created snapshots with w sizes 3-hour, 6-hour, 12-hour, and lastly 24-hour. For Haggle Infocomm, the w sizes are determined as every minute, every five minutes, every fifteen minutes, and every hour. Finally for Sabancı Wi-Fi logs, candidate w sizes are five, ten, fifteen, thirty and sixty minutes intervals.

4. Results

In the following sections, we first show and interpret the results of baselines and our compression ratios together to find the best w among candidate intervals. Then, we evaluate the results of the proposed aggregation algorithm by interpreting the topological properties of the network, namely diameter, average degree, average distance, or node number.

4.1. Choosing The Best Window Size

The best window sizes for four data sets are determined by TWIN, stability, average link similarity, and link and node compression ratios. For TWIN, we adopt the network diameter as it is done in [29]. TWIN results are displayed in Fig. 2. In the figure, we show the variance and compression ratio of the diameter for different w. The best w is the one whose variance and compression ratio are the closest to each other. Accordingly, for Enron, the best w is 7 days. For Haggle Infocomm, in fact, variance and compression ratio coincide on both 5 and 60 minutes. For reality mining, the best w is 3 hours.

These three data sets were previously studied by Sulo et al. [29] by applying TWIN. We have found similar results with them, except for reality mining. In a previous application, Sulo et al. found 6 hours for the best w [29]. In fact, the reality mining network can be extracted in two different ways. One of them is putting a link between two devices that see each other at a given period of Bluetooth scan. The other way is to put a link between two devices that scan the same devices at a given period. The former network is denser and includes noisy links, while the latter takes into account only the main devices that participate in the experiment. We extract the snapshots according to the former method. This can be the reason why our implementation results are different from the ones presented in [29]. For other data sets, the reliability of our TWIN implementation and experimental set up has been validated. For Sabanci University Wi-Fi logs, which is a new experimental data set for this problem, the best window size that TWIN finds is 15 minutes.

In Figure 3, we show both stability scores and average link similarity that are found for different data sets. Stability is a specific version of average link similarity weighted on the size of the minimum link set (please refer to Section 2.1 for details). These data sets have never been studied before by stability. According to both stability and average link similarity, the most stable window size for Enron is 15 days. For Haggle Infocomm,

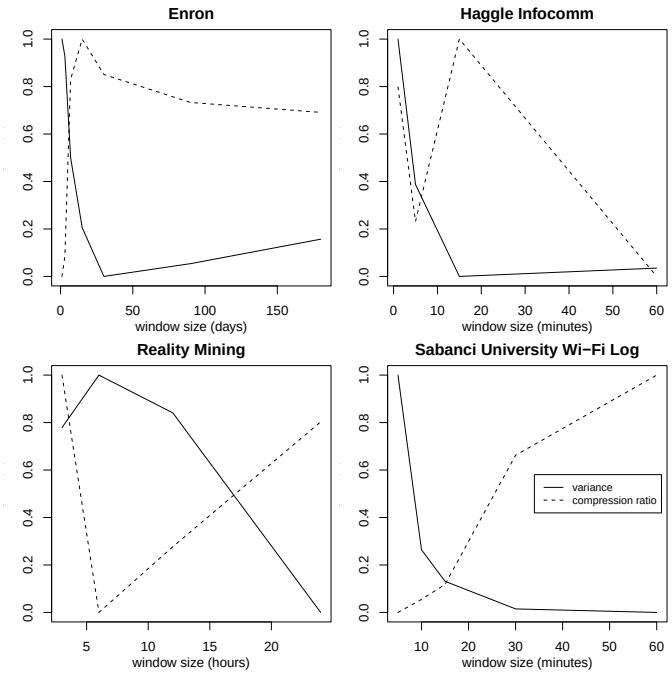


Fig. 2. TWIN results calculated by variance and compression ratio of diameter for Enron, Haggle Infocomm, Reality Mining and Sabanci University Wi-Fi logs. The most proper window sizes are 7 days, 5 minutes, 3 hours and 15 minutes respectively

the two metrics demonstrate opposite behaviour. The higher the duration, the higher the stability, but the lower the average link similarity. This can be due to not only the connections but also the system actors changing dynamically. The best w is 5 minute for average link similarity and 60 minutes for stability. For reality mining, both metrics show a similar trend. The best w is 3 hours. For Sabanci University Wi-Fi logs, 10 minutes seems to be a critical period. Two metrics show similar trends for longer periods than 10 minutes. But for lower periods, stability takes a higher value while average link similarity is lower. We have indicated that the data is collected every 10 minutes in the section 3. That is why this period is already significant for the breaking point of different logging moments. We have found out that the best w is 5 minutes for stability and 10 minutes for average link similarity.

The results of node and link compression are shown in Fig. 4. Overall results are listed in table 2. Accordingly, for Enron, we find 15 days to be the best window size. Both node and link compression results agree on this value. Our result is the same as stability and average link similarity but different from the TWIN result. That's why we examine in detail the diameter signal of Enron when $w = 7$ and 15 days. Those signals are shown in Fig. 5. The compression ratio found by TWIN for diameter is the same at 7-day and 15-day intervals. Because the variance of 7 days in the calculations is greater than that of 15 days, TWIN determined that 7 days is the best w. However, as we mentioned previously, topological properties and variance are not scale invariant.

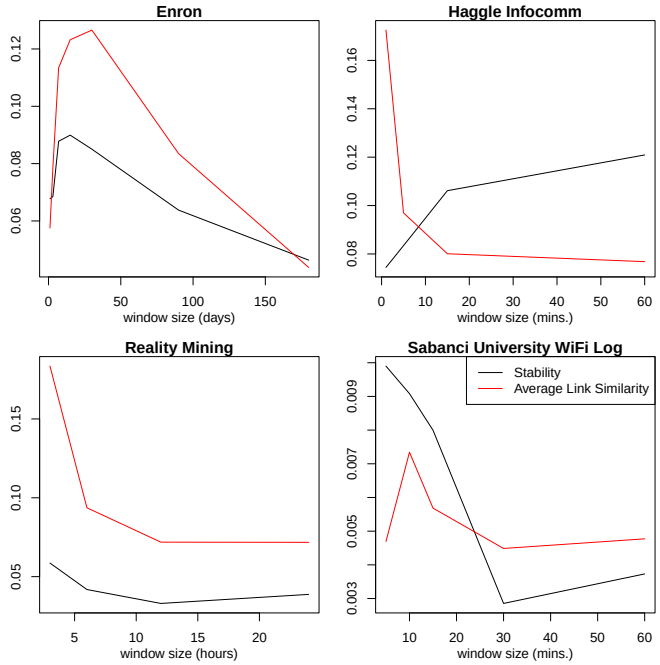


Fig. 3. Stability scores calculated by Equation 5. The most proper window sizes are 15 days, 60 minutes, 3 hours and 5 minutes respectively

Topological properties can have different scores reflecting the same information for two snapshots, or vice versa. In our detailed analysis, we saw that the diameter scores found for these two w were different. When we looked at the normalised standard deviation with the mean over the obtained values instead of simple variance, we found that the variation of the 15-day interval was higher. Hence, variance, which is not a normalised measure, can be misleading for explaining the noise of generated signals.

In order to get deeper in the analysis for this hypothesis, we examined the diameter signals for the two w in detail. The results are depicted in Fig. 5. Some of the important events related to the Enron Company's collapse are marked with arrows and with different colours in the Fig. 5. Accordingly, the first event, Event1, in the signal part that coincides with the date of December 2001, is the period when Enron announced bankruptcy. Event1 is hidden in the plot of $w = 7$ because it is both noisy and does not reflect the significance of the event with a critical peak. Contrarily, we can see important details with less noise but in sufficient detail in the plot of $w = 15$. The critical diameter peak is also remarkable in the section corresponding to the bankruptcy date.

More information can be retrieved from Enron signals. For example, Event2 is the period when George Bush named Kenneth Lay, who is Enron's chairman and contributed more than $290,000 to George Bush's election campaign, as an adviser to his presidential transitional team in January 2000. We catch this event in the plot of $w = 15$ with a high peak. However, it is hidden among the noise of the signal and cannot be caught in the plot of $w = 7$. Similarly, around February and March 2002, the hearing started before

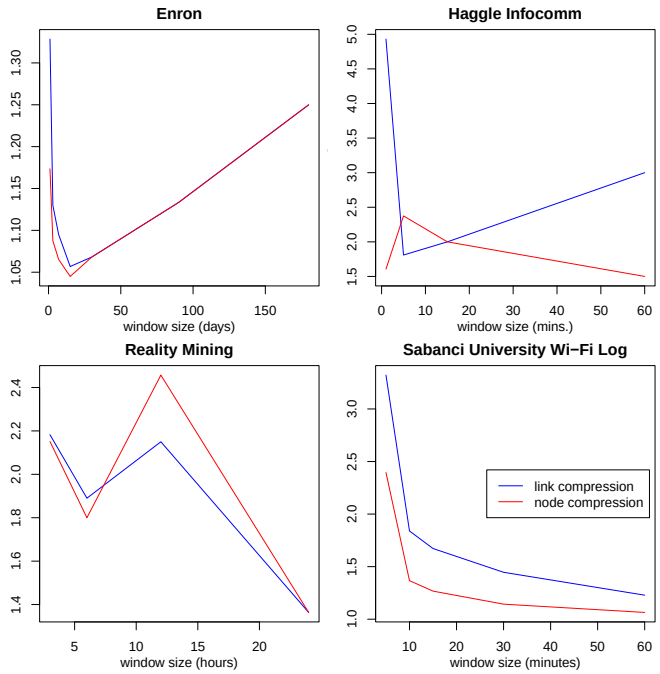


Fig. 4. Link and Node Compression results for Enron, Haggle Infocomm, Reality Mining and Sabanci University Wi-Fi logs. The most proper window sizes are 15 days, 5 minutes, 6 hours and 60 minutes respectively

Table 2. The best window sizes for four data sets according to different methods

	Enron	**Haggle Infocomm**	**Reality Mining**	**Sabanci University Wi-Fi logs**
TWIN	7 days	5 mins./60 mins.	3 hours	15 mins.
Stability	15 days	60 mins.	3 hours	5 mins.
Average Link Similarity	15 days	5 mins.	3 hours	10 mins.
Node Compression Ratio	15 days	60 mins.	6 hours/24 hours	60 mins.
Link Compression Ratio	15 days	5 mins.	6 hours/24 hours	60 mins.

the Senate. This event, Event3, is clearly visible in the plot of $w = 15$. However, it is represented by multiple snapshots in the plot of $w = 7$. Moreover, on June 15, 2002, Arthur Andersen was convicted. This event, Event4, is captured by both $w = 7$ and $w = 15$. We could discern many significant events from the company's routine, such as the resignation of the chairman and chief executive of Enron at the peak points of the signal of $w = 15$. When we look at the overall picture, the window size found by node and link compression seems to be more informative in terms of reflecting real-world phenomena for this system than the one that TWIN found.

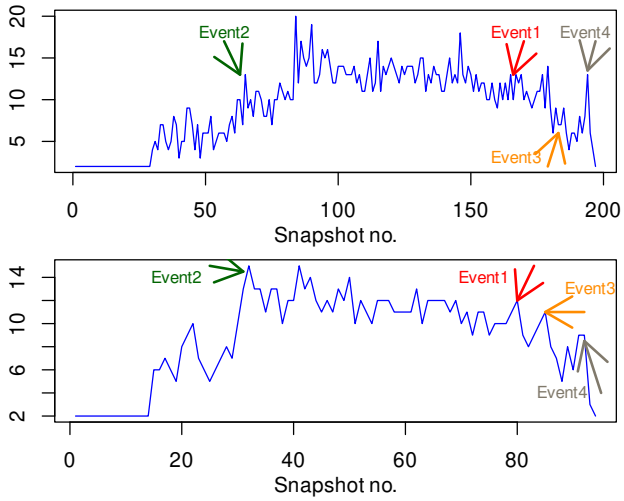


Fig. 5. Top and bottom time series are Enron diameter scores for $w = 7$ and $w = 15$ days respectively. The biggest bankruptcy in US history happened at December 2001. The date is indicated with red arrow in both plots

For Haggle Infocomm, link compression finds the best result for w as 5 minutes, but node compression has the opposite behaviour (see Fig. 4). In fact, node similarity represents the stability of the actors in the system. If the system is changing slowly or the studied w sizes are too small, it can be expected that the nodes of the system do not change considerably. Nevertheless, link similarity represents the stability of the connection between system actors. Although the system actors stay the same, their connections can change. Let us remind you that Haggle Infocomm is a set of controlled experiments including Bluetooth connections in a predefined environment. The participants of this conference did not change considerably, whereas their connections were actively changing during the conference. Thus, in this system, we take into account the results of link compression instead of node compression. As a result, we obtain the same result with TWIN and average link similarity. Our node compression result is the same as stability.

For Reality Mining, the lowest values of compression scores are obtained when w is 6 and 24 hours. Surprisingly, these are two different suggested w values, as explained in two separate articles, [29] and [11]. If we want to capture sudden changes occurring daily, we extract the network with $w = 6$. But if we want to observe the system with its macro changes and evaluate them, we use $w = 24$. The studied baselines, TWIN, stability, and average link similarity, result in different ways from our proposals. We have found 3 hours to be the best w according to these baselines. But the detailed topological analysis reveals that, as in the case of Enron, the noise of the $w = 3$ signals obscures the important events. We show the change of link number in the snapshots of $w = 3$, $w = 6$ and $w = 24$ in the figure 6 for the first two months of the data set. There is a slight difference between $w = 3$ and $w = 6$ signals, while the signal of $w = 24$ shows only weekly changes. We can distinguish the daily circadian rhythm from the weekday

activity ups and downs by a cyclic signal in both $w = 3$ and $w = 6$. Moreover, the lower activity of the weekend is also discernible. When comparing $w = 3$ with $w = 6$, there is no significant difference, but the signal is more noisy when $w = 3$. For $w = 3$ and 6, the normalised standard deviation is 1.46 and 1.32, respectively. Accordingly, when w is 3 hours, extracted snapshots become noisy and seem to include the same information with a higher window size, $w = 6$. Among 6 and 24 hours, it seems like 6 hours is at an optimal granularity level, while 24 hours smooths too much of the daily activity. Therefore our proposed compression ratios suggest better window sizes that the baseline methods.

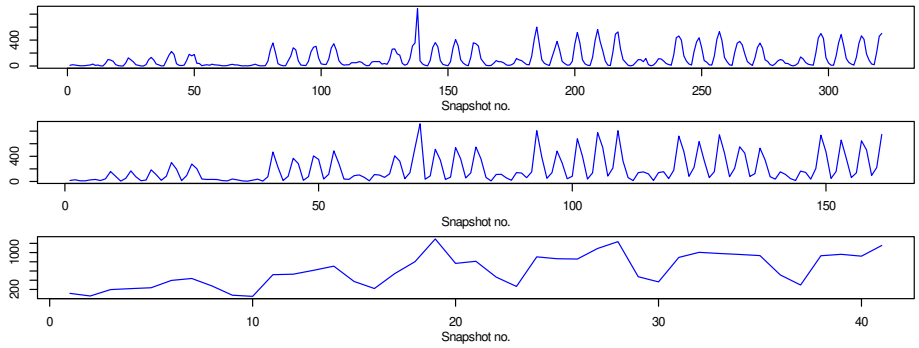


Fig. 6. Top, centre and bottom time series are MIT Reality Mining link number scores for first two months of $w = 3$, $w = 6$ and $w = 24$ hours respectively

When the compression ratio of Sabancı University Wi-Fi logs is analysed logs', an exponential decay with an increase of w is observed. Thus, the lowest values of both compression ratios are obtained for $w = 60$ minutes. Because this is different from baseline results, we study the signals in depth. The daily circadian rhythms are clearly observed in the average degree signals of Sabanci University Wi-Fi Logs for $w = 5$, $w = 10$, $w = 15$ and 60 minutes intervals (see Fig. 7). We can also notice the effect of lower activity at the weekend on both signals. The first cycle and the seventh and eighth cycles in the plots belong to the 4th, 9th, and 10th of December 2016, Sunday, Saturday, and Sunday respectively. The rest of the cycles reflect the daily activity of campus life. Accordingly, the activity increases from the morning till afternoon, afterwards it starts to decrease. Moreover, during the daytime, there are some periods when the activity peaks occur. The signal of 5, 10, and 15 minutes seems to be too noisy to reflect the daily picks. The normalised standard deviation of average degree signals is also given in the figures. As the signal plots show, the noise level of all signals lower than 60 minutes is higher than the one of 60 minutes. Similar to the results obtained for the previous data sets, our proposed method reflects more realistic, less noisy and easily interpretable results than the basic approaches for the Sabancı University Wi-Fi logs data set.

4.2. Qualitative Performance of Window Aggregation Algorithm

Once we determined the best w as explained in details in the previous section, we applied the window aggregation algorithm in order to obtain a proper dynamic network

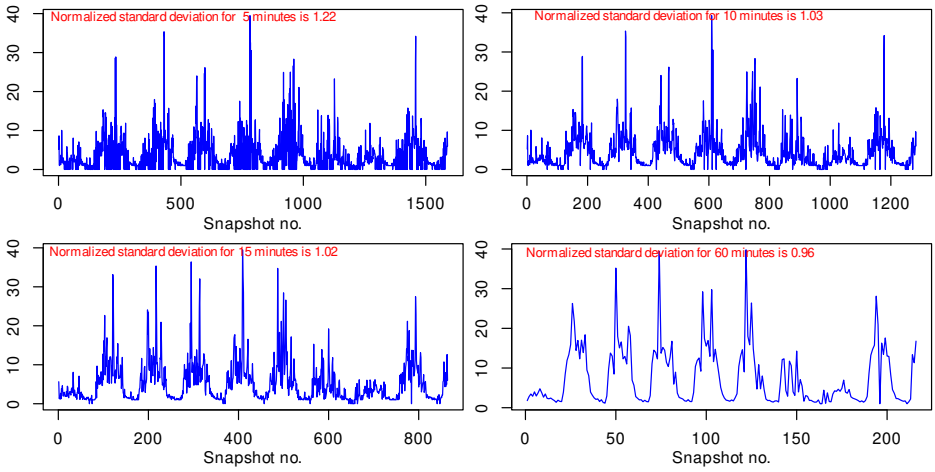


Fig. 7. Top-left, top-right, bottom-left and bottom-right time series are Sabanci University Wi-Fi logs average degree scores for $w = 5$, $w = 10$, $w = 15$ and $w = 60$ minutes respectively. The related normalised standard deviation is written with red text on each plot

representation for the data sets. We evaluate the resulting networks based on the similarity of consecutive snapshots as well as their topology. On one hand, for Enron and Reality Mining data sets, node and link similarity signals of snapshots with the constant w are difficult to analyse since there are no characteristic changes that give meaningful information about the system. as explained in details in the previous section.

On the other hand, we detect meaningful changes in node similarity signals of aggregated snapshots. These peak values were latent in the signal of constant snapshots. They were undiscernable in the flat stationary signals. Therefore, the aggregation strategy allows us to detect instantaneous changes in the system. Moreover, we could detect peak values, which are the points where the system shows a considerable change. These remarks are also validated with the Haggle data set.

Node and link similarity signals for both constant snapshots and aggregated snapshots of Haggle data sets are shown in Fig. 8. The selected window size for this analysis is 5 minutes. If we take a look at node similarity plots for constant snapshots, the signals show a circadian rhythm, but they are too noisy. This data set was collected for four days. We clearly observe daily cycles between day and night in constant window usage with the periodic increase and decrease in the signals. These types of circadian rhythms are common and explained in detail in [2]. Obtaining circadian signals does not add/contribute much to understanding of the studied system. However, although the signals generated by aggregated snapshots have no distinct trend, one can detect peak values and important changes in specific time points. Because circadian periods are aggregated and represented with fewer snapshots, we observe only important changes in those signals. The topology of these snapshots supports our interpretation.

We obtain similar circadian rhythms for Haggle Infocomm as well. In Fig. 9, diameter and average distance signals for constant (top layout) and aggregated (bottom layout)

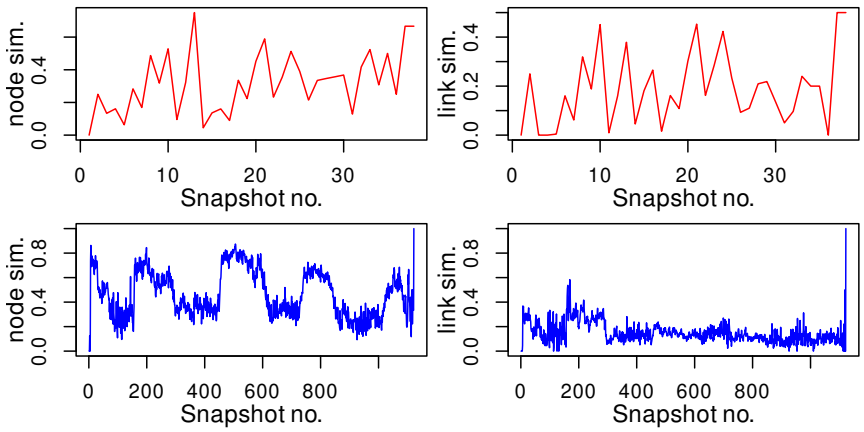


Fig. 8. Node and Link Similarity signals of Haggle snapshots extracted for $w = 5$ minutes from left to right, respectively. The plots of the snapshots extracted with aggregated windows and constant window are given in top and bottom layout, respectively

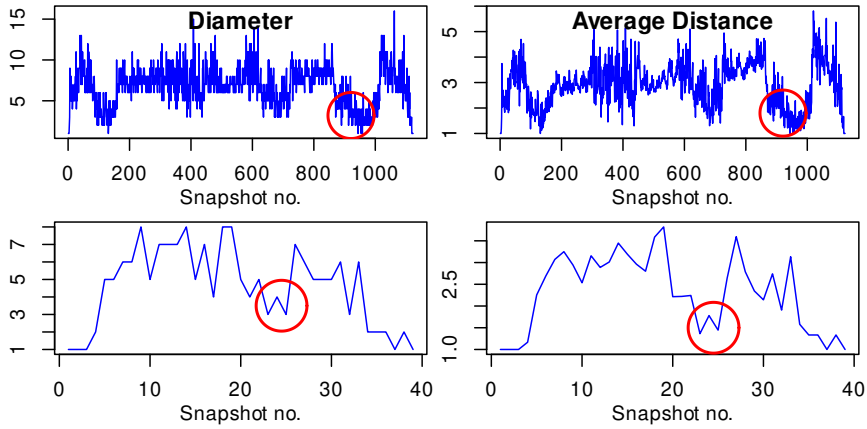


Fig. 9. Diameter and Average Distance Signals for $w = 5$ minutes for Haggle Infocomm. The plots of the snapshots extracted with constant window and aggregated windows are given in top and bottom layout, respectively. The red circles correspond to the same period of critical decrease for all plots

windows for the Haggle Infocomm data set are shown. The red circled zones of both plots correspond to the same period, the last night of the conference. Some of the participants have already left. That's why the number of nodes and links decreases, and the diameter and average distance get shorter. The aggregation process merges many snapshots before that period because the system repeats similar activity. However, the decline in this period is greater than in the previous one. As a result, they are represented as separate snapshots. The final night of the conference appears to be similar to other nights when we extract by

constant window size. We cannot understand its difference because of having too many repetitive snapshots, i.e., a noisy signal.

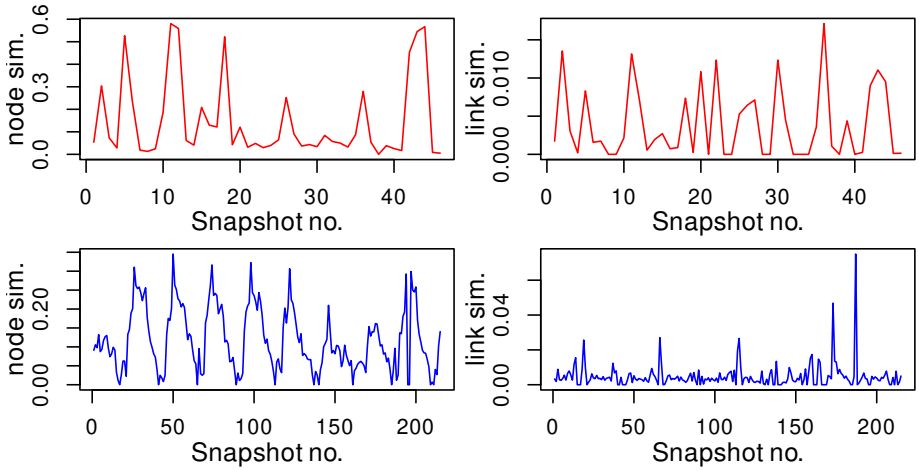


Fig. 10. Node and Link Similarity signals of Sabanci University Wi-Fi logs snapshots extracted for $w = 60$ minutes from left to right, respectively. The plots of the snapshots extracted with aggregated windows and constant window are given in top and bottom layout, respectively

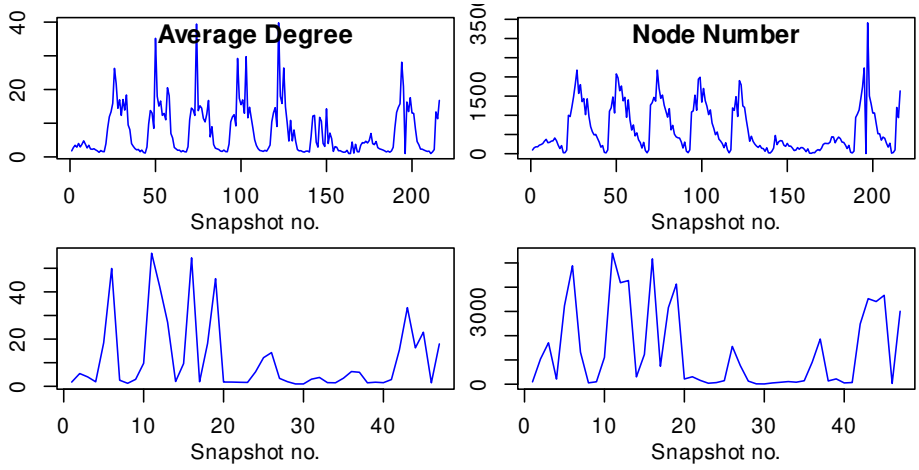


Fig. 11. Average Degree and Node Number Signals for $w = 60$ minutes for Sabanci University Wi-Fi logs. The plots of the snapshots extracted with constant window and aggregated windows are given in top and bottom layout, respectively

Another data set that reflects the circadian rhythms of the studied system is the Sabanci University Wi-Fi logs. Differently from Haggle experiments, Sabanci University Campus is an active area in which the members of the system change dynamically. Therefore, not only the link but also the node structure changes during the day and also the week. We observe the effect of the aggregation algorithm in Fig. 10 and 11. The similarities of consecutive aggregated snapshots are shown in the top plot of Fig. 10. Accordingly, especially from node similarity, we can clearly see the circadian rhythm, the consecutive increase and decrease of activity from day to night and, moreover, the lower activity of weekends. This fact is also observed in network topology (see Fig. 11). As in our previous comment about Sabanci University Wi-Fi logs, the campus population declines on weekends, only some students staying in the university dormitory connect to the Wi-Fi access points. However, during the week days, both students and university staff go on and off campus. They move around the campus by connecting to the different Wi-Fi access points at different locations, which in turn creates more activity on weekdays.

Comparing two dynamic networks through their signals; the one extracted by an aggregation algorithm and the one with constant window size, we notice clearly that the aggregation algorithm allows us to model the system by keeping its essential properties, such as the life cycle periods of the campus, with clearer signals. In Section 4.1, we saw that the node and link compression ratio let the modeller extract a dynamic network model with less noise. The aggregation algorithm seems to refine the modelling. This was a regular period when the campus exhibited its usual activities. That's why; we do not observe any peak or dramatic change except for circadian rhythms. The most remarkable result of the aggregation algorithm for Sabanci University Wi-Fi logs, however, is that it allows the extraction of the final dynamic network with features reflecting almost no noise, while still retaining the dynamics and properties of the studied system.

4.3. Discussion

The experimental results reveals that both novel compression ratios and aggregation algorithm reduce the noise in the data. They ensure the extraction of clearer, dynamic network models whose analysis is easier for knowledge discovery. Data sets collected from telecommunications networks such as e-mail exchange, Facebook and Twitter are already convenient for network analysis. For them, direct relationships between person A and person B can be represented with network links. Therefore these data sets cause relatively less noise or less challenge for dynamic network modelling. On the other hand, proximity data sets, such as Haggle Infocomm or Sabanci University Wi-Fi Logs, do not directly reflect social relationships, but they may help us discover common behaviours among people in the same environment. Stopczynski demonstrates that the physical proximity of people can be inferred by their connection to the same Wi-Fi access points within a sufficiently small time window [28]. However, there are many problems with limiting the handling of raw logs. Some of them are listed in [17] as first because it is noisy data compared to other location data sources such as GPS. Second, the data might contain some misleading information. However, our proposed aggregation algorithm helps us to eliminate the noise when using optimal w. Consequently, it might be useful for discovering the behaviour analysis of system actors in further analysis.

Although similarity-based compression ratios and aggregation algorithm result in less noisy and more informative dynamic network models, they can be criticised because they

consider only local information. The methodology proposed here is completely based on the Jaccard similarity of node-or link-sets of consecutive snapshots. Accordingly, the topological role or position of the nodes or links is not privileged. For example, the disappearance of one critical link in the role of a bridge and the disappearance of one simple link return the same link similarity result, although their effect on the system is completely different. We measure the role of the nodes, or links, via their topological properties. However, it can be possible to neglect the system's emerging properties by using only local comparison of nodes or link sets. Because such an analysis is outside the scope of this work, we do not consider this problem. For further information please refer to our previous study [22], where the authors showed that the use of similarity instead of topological properties allows one to extract more informative models.

5. Conclusion

This research focuses on the accurate modelling of complex systems, such as interacting objects in the form of dynamic networks that evolve in real time. It solves both of the model's key problems: first, selecting the ideal temporal window size, w, and second, extracting the appropriate dynamic network using w. It provides two novel dynamic network compression metrics based on the similarity ratios of sequential snapshots to find w, as well as a window aggregation approach using the previously determined w. According to experiments on four data sets, compression ratios can extract more noise-free and informative networks than baseline approaches. Furthermore, the aggregation method has managed to lower noise levels even further without compromising the system's overall and crucial features.

Three of the studied data sets consist of Bluetooth or Wi-Fi connection data, which may contain the proximity information of the people in the system. The analysis of these data sets with basic methods is quite challenging, as they are noisy. However, people's behavioural patterns are hidden behind them. The most important principal for these data sets is that a noise-reduced analysis model could be built by using proposed compression ratios and aggregation algorithms. In Sabanci University Wi-Fi log data, for example, we were able to observe system-specific circadian results such as campus life cycle and weekday-to-weekend activity differences with noise-reduced signals. Or, in Haggle Infocomm data, we were able to observe a decrease in activity on the last day of the conference, which is a detail that was not observed before the noise was eliminated by our proposals. Apart from all these three sets, we were able to capture the important dates of the collapse of the company in the Enron data set in the networks obtained by our approach. Those event details were not observed in the noisy modelling.

All these results show us that we can make a correct dynamic network model, especially in data sets that contain complex structures. The accuracy of the extracted models should be supported by further analysis like link prediction, community detection, knit group finding, or behaviour prediction in the next steps. This work can be extended in several ways. For example, the use of candidate w when determining the best w still requires system expertise. This can be improved over the proposed compression ratios. These ratios, in their current form, compress the snapshots against an analytically calculated upper limit of similarity. This limit gives us the lowest possible similarity. However, we have not yet established the highest possible similarity analytically. More

precisely, we can decide that two consecutive snapshots are similar but cannot decide that they are dissimilar. This way, we can further refine the compression ratio by fixing a second limit. This allows us to determine more accurate time windows. Thus, as a result, we can achieve the best w by optimising w value according to lower and upper limits without using candidate w. The work proposed here is based on the local measurement of Jaccard similarity and does not take into account the topological position of links or nodes. This can cause neglect of the emerging and non-linear properties of the studied system. Another perspective of this work could be to measure the weighted Jaccard in a way to quantify the topological importance of the nodes or links for considering emerging properties. Furthermore, this work can be adapted and tested on the systems from different domains that are evolving at different speeds. Thus, the effect of the system's being fast or slow on the proposed methods can be examined.

Acknowledgments. We thank Prof. Dr. Selim Balcisoy from Computer Engineering Department of Sabanci University for providing Sabanci University Wi-Fi Log dataset. This article is partially supported by Galatasaray University Research Fund (BAP) within the scope of project number fba-2021-1063, and titled "Niteliklendirilmiş çift yönlü ağlarda bağlantı tahmini ile öneri sistemleri geliştirilmesi".

References

1. Aggarwal, C., Subbian, K.: Evolutionary network analysis: A survey. ACM Computing Surveys 47(1), 10:1–10:36 (2014)
2. Aledavood, T., López, E., Roberts, S.G.B., Reed-Tsochas, F., Moro, E., Dunbar, R.I.M., Saramäki, J.: Daily rhythms in mobile telephone communication. PLOS ONE 10(9), 1–14 (09 2015), https://doi.org/10.1371/journal.pone.0138098
3. Blonder, B., Wey, T.W., Dornhaus, A., James, R., Sih, A.: Temporal dynamics and network analysis. Methods in Ecology and Evolution 3(6), 958–972 (2012)
4. Caceres, R.S., Berger-Wolf, T., Grossman, R.: Temporal scale of processes in dynamic networks. In: 2011 IEEE 11th International Conference on Data Mining Workshops. pp. 925–932 (Dec 2011)
5. Cazabet, R.: Data compression to choose a proper dynamic network representation (2020)
6. Chiappori, A., Cazabet, R.: Quantitative evaluation of snapshot graphs for the analysis of temporal networks (2021)
7. Clauset, A., Eagle, N.: Persistence and periodicity in a dynamic proximity network. CoRR abs/1211.7343 (2012)
8. Dakiche, N., Tayeb, F.B., Slimani, Y., Benatchba, K.: Sensitive analysis of timeframe type and size impact on community evolution prediction. In: 2018 IEEE International Conference on Fuzzy Systems. pp. 1–8 (2018)
9. Donnat, C., Holmes, S.: Tracking network dynamics: A survey using graph distances. The Annals of Applied Statistics 12(2), 971 – 1012 (2018), https://doi.org/10.1214/18-AOAS1176
10. Eagle, N., Pentland, A.S., Lazer, D.: Inferring social network structure using mobile phone data. In: PROCEEDINGS OF NATIONAL ACADEMY OF SCIENCES. pp. 127–136 (2009)
11. Eagle, N., (Sandy) Pentland, A.: Reality mining: Sensing complex social systems. Personal Ubiquitous Comput. 10(4), 255–268 (Mar 2006)
12. Fish, B., Caceres, R.S.: Handling oversampling in dynamic networks using link prediction. In: Machine Learning and Knowledge Discovery in Databases. pp. 671–686. Springer International Publishing (2015)

13. Fish, B., Caceres, R.S.: A supervised approach to time scale detection in dynamic networks. CoRR abs/1702.07752 (2017)
14. Holme, P., Saramäki, J.: Temporal networks. Physics Reports 519(3), 97–125 (2012)
15. Jo, H.H., Karsai, M., Kertész, J., Kaski, K.: Circadian pattern and burstiness in mobile phone communication. New Journal of Physics 14(1), 013055 (jan 2012), https://doi.org/10.1088%2F1367-2630%2F14%2F1%2F013055
16. K. Darst, R., Granell, C., Arenas, A., Gomez, S., Saramäki, J., Fortunato, S.: Detection of timescales in evolving complex systems. Scientific Reports 6 (04 2016)
17. Kjærgaard, M.B., Nurmi, P.: Challenges for social sensing using wifi signals. In: Proceedings of the 1st ACM Workshop on Mobile Systems for Computational Social Science. pp. 17–21. MCSS '12, ACM, New York, NY, USA (2012)
18. Krings, G., Karsai, M., Bernhardsson, S., Blondel, V.D., Saramäki, J.: Effects of time window size and placement on the structure of an aggregated communication network. EPJ Data Science 1(1), 4 (May 2012)
19. Kuhn, F., Oshman, R.: Dynamic networks: Models and algorithms. ACM SIGACT News 42(1), 82–96 (2011)
20. Léo, Y., Crespelle, C., Fleury, E.: Non-altering time scales for aggregation of dynamic networks into series of graphs. In: Proceedings of the 11th ACM Conference on Emerging Networking Experiments and Technologies. CoNEXT '15, Association for Computing Machinery, New York, NY, USA (2015), https://doi.org/10.1145/2716281.2836114
21. Medo, M., Zeng, A., Zhang, Y., Mariani, M.S.: Optimal timescale of community detection in growing networks. CoRR abs/1809.04943 (2018)
22. Orman, G.K., Türe, N., Balcisoy, S., Boz, H.A.: Finding proper time intervals for dynamic network extraction. Journal of Statistical Mechanics: Theory and Experiment 2021(3), 033414 (mar 2021), https://doi.org/10.1088/1742-5468/abed45
23. Real, R., Vargas, J.M.: The probabilistic basis of jaccard's index of similarity. Systematic Biology, JSTOR 45(3), 380–385 (1996), www.jstor.org/stable/2413572
24. Rossi, R.A., Gallagher, B., Neville, J., Henderson, K.: Modeling dynamic behavior in large evolving graphs. In: Proceedings of the Sixth ACM International Conference on WSDM. pp. 667–676 (2013)
25. Song, C., Qu, Z., Blumm, N., Barabási, A.L.: Science 327(5968), 1018–1021 (2010)
26. Soundarajan, S., Tamersoy, A., Khalil, E.B., Eliassi-Rad, T., Chau, D.H., Gallagher, B., Roundy, K.: Generating graph snapshots from streaming edge data. In: Proceedings of the 25th International Conference Companion on WWW. pp. 109–110 (2016), https://doi.org/10.1145/2872518.2889398
27. Spiliopoulou, M.: Evolution in social networks: A survey. In: Social Network Data Analytics, chap. 6, pp. 149–175. Springer (2011)
28. Stopczynski, A., Sekara, V., Sapiezynski, P., Cuttone, A., Madsen, M.M., Larsen, J.E., Lehmann, S.: Measuring large-scale social networks with high resolution. PLOS ONE 9(4), 1–24 (04 2014)
29. Sulo, R., Berger-Wolf, T., Grossman, R.: Meaningful selection of temporal resolution for dynamic networks. In: Proceedings of the Eighth Workshop on Mining and Learning with Graphs. pp. 127–136. MLG '10, ACM, New York, NY, USA (2010)
30. Uddin, S., Choudhury, N., M. Farhad, S., Rahman, M.: The optimal window size for analyzing longitudinal networks. Scientific Reports 7 (12 2017)
31. Çolak, S., Orman, G.: Aggregating Time Windows for Dynamic Network Extraction (2021)

Günce Keziban Orman has a PhD in Computer Science from INSA de Lyon. She is working as a full-time assistant professor at Galatasaray University, Istanbul, Turkey.

She works on complex network analysis in general and accurate network modelling, community detection, and link prediction in particular.

Received: September 29, 2021; Accepted: December 26, 2021.

Computer Science and Information Systems 19(2):1047–1073 https://doi.org/10.2298/CSIS211123013L

A Low-Cost AR Training System for Manual Assembly Operations*

Traian Lavric[1,2], Emmanuel Bricard[1], Marius Preda[2], and Titus Zaharia[2]

[1] IP Paris - Telecom SudParis, 91011, Évry, France
{traian.lavric, marius.preda, titus.zaharia}@telecom-sudparis.eu
[2] ELM LEBLANC SAS, 937400, Drancy, France
{traian.lavric, emmanuel.bricard}@fr.bosch.com

Abstract. This research work proposes an AR training system adapted to industry, designed by considering key challenges identified during a long-term case study conducted in a boiler-manufacturing factory. The proposed system relies on low-cost visual assets (i.e., text, image, video, and predefined auxiliary content) and requires solely a head-mounted display (HMD) device (i.e., Hololens 2) for both authoring and training. We evaluate our proposal in a real-world use case by conducting a field study and two field experiments, involving 5 assembly workstations and 30 participants divided into 2 groups: (i) low-cost group (G-LA) and (ii) computer-aided design (CAD)-based group (G-CAD). The most significant findings are as follows. The error rate of 2.2% reported by G-LA during the first assembly cycle (WEC) suggests that low-cost visual assets are sufficient for effectively delivering manual assembly expertise via AR to novice workers. Our comparative evaluation shows that CAD-based AR instructions lead to faster assembly (-7%, -18% and -24% over 3 assembly cycles) but persuade lower user attentiveness, eventually leading to higher error rates (+38% during the WEC). The overall decrease of the instructions reading time by 47% and by 35% in the 2nd and 3rd assembly cycles, respectively, suggest that participants become less dependent on the AR work instructions rapidly. By considering these findings, we question the worthiness of authoring CAD-based AR work instructions in similar industrial use cases.

Keywords: augmented reality, training, content authoring, work instructions, assembly, user study, industry 4.0.

1. Introduction

The industrial revolution also known under the label of Industry 4.0 provides a set of enabling technologies that support the development of individualized products in a cost-effective manner [41]. Augmented Reality (AR) is one of the key technologies that have demonstrated its benefits as a knowledge-sharing tool, among other applications in a variety of domains including education, medicine, tourism and entertainment [1][5]. Studies show that AR training systems can be more efficient in terms of task completion time and error rates when compared to classical training procedures (i.e., paper instructions) [44][2][40][10][7]. Although AR has been investigated as a guidance tool for manufacturing process since more than two decades [4], only recently, technological advancements

* The present paper is an extended and revised version of our preliminary conference report that was presented in INISTA 2021 [16]. This paper significantly expands the evaluation of the proposed AR training method.

enabled a resurgence of the AR use. However, despite the exponential progress that AR has experienced in recent years, no significant breakthrough can be noted in the industrial environment, due to various challenges [25][26]. AR systems have been mostly designed and evaluated in controlled environments, under laboratory settings, as recent surveys show [29][6]. Palmarini *et al.* [37] claimed that AR technology is not sufficiently mature for complying with strong industrial requirements such as robustness and reliability. Another recent study, conducted by Masood and Egger [26], identified and classified AR challenges into three main categories: technology, organization and environment and uncovered a gap between academic and industrial challenges. The authors suggested that field studies must be conducted in order to ensure the successful implementation of AR systems in industrial sectors.

We aimed to address these recommendations by elaborating an AR training solution for a concrete use case, a boiler-manufacturing factory. To this purpose, we conducted a long-term case study for obtaining a comprehensive picture of the needs and requirements, from both technical and organizational perspectives, that an AR training system should address, to be adopted in such context. The key success factors identified during our case study are effectiveness and viability. A summary of the most significant challenges that an AR training system should address, to be considered for adoption in the considered use case are safety, user acceptance, viability, technical setup, existing digital resources, assembly environment and process. A detailed description of our case study and its findings are presented in [16].

This work has an industrial focus; however, it explores relevant AR-related research topics identified by Kim *et al.* [14], including interaction techniques, user interfaces (UI), AR applications, evaluation, AR authoring, visualization and multimodal AR. Additionally, it addresses AR assembly concerns identified by Wang *et al.* [47], including time-consuming authoring procedures and appropriate guidance for complex, multi-step assembly tasks. Finally, it tries to answer a research question inquiring optimal ways for conveying instructions in Industrial Augmented Reality (IAR) [9]. We adopted therefore a human-centered design (HCD) approach to provide an intuitive, hands-free AR training system adapted to the shop floor environment, by addressing some of the most relevant industrial concerns identified during our case study and in the literature as well [6][21][27][39][35]. We evaluated the proposed AR training method, particularly the conveyance of the AR step-by-step instructions by conducting two field experiments. The first [16], a preliminary one that involved 12 participants, aimed at assessing the effectiveness and usability of our proposed low-cost AR training method. The second [17], an extension of the first, involved 20 additional participants and aimed at assessing the worthiness of using CAD data for conveying manual assembly expertise via AR. This paper presents a comprehensive overview of the two field experiments and discusses unpublished preliminary data in respect to the proposed AR authoring method, collected during a field study. The overall reported evaluation results collected from the considered experiments suggest that capturing and conveying expert knowledge via AR by using uniquely low-cost spatially registered visual assets is potentially the most efficient and viable option until the creation, manipulation and storage of CAD data and animations become more convenient, especially in industrial sectors.

The rest of this paper is organized as follows. Section 2 describes our proposed AR training method. Section 3 presents the technical implementation of the system. Section 4

describes the field experiments. We discuss the most significant findings in Section 5. A summary of the conclusions is presented in Section 6. Finally, limitations and suggestions for future work are discussed in Section 7.

2. Proposed Method

In this section, we justify the most relevant choices on which our proposal relies (see Section 2.1), we discuss the main design principles of our approach (see Section 2.2) and finally, we elaborate the proposed methodology, for both training and authoring (see Section 2.3). A summary of the main concerns that our proposed methodology aims to address, are further listed:

- **Content**: the AR system should not rely on existing digital data.
- **User**: the AR system should be adapted to shop floor personnel, particularly to assembly line experts and novice workers.
- **Environment**: the AR system should be hands-free, usable and effective, independently on the assembly environment.

2.1. AR Device, Visual Assets and Spatial Registration

To address the aforementioned concerns from a hardware perspective, our findings indicate that the best compromise is using cable-less HMD AR devices. Handheld devices (i.e., smartphones) do not answer the hands-free requirement while spatial augmented reality (SAR) systems [28][46] are not considered viable for the considered manufacturing context, shows our study. The methodology and implementation of the proposed AR training system relies therefore on the state-of-the-art AR device, Microsoft® HoloLens 2 [30], further referred to as Hololens 2.

The second most important aspect is represented by the way the assembly information is conveyed via AR. Literature shows that digital assets used to convey information in AR include text, audio, static 2D/3D and dynamic 2D/3D [20]. The visual ones are classified as text, sign/symbol, image/picture, video, drawing, 3D model and animations [9][20]. However, as identified in a recent study [9], there is no agreement in the literature regarding optimal ways of conveying instructions via AR. Tainaka *et al.* [43] empirically observed however that low-cost visual assets provide satisfactory results in conveying most manual assembly operations. Lee *et al.* [19] demonstrated the potential of first-person view (FPV) videos for conveying task instructions. In addition, we remark a potentially significant advantage of low-cost assets: unlike CAD models, these can be captured by state-of-the-art AR devices (i.e., Hololens 2), in-situ, as part of the AR authoring procedure itself. The authoring of the AR instructions is therefore not limited by existing digital content, preparation or post-processing steps, as proposed by commercial AR tools like Vuforia Expert Capture [38] and Microsoft Dynamics 365 Guides [33], further referred to as Guides. A summary of the most relevant concerns related to the usage of CAD models in AR, identified during our informal experiments are availability [21] and preparation, positioning during the authoring, occlusion, and real time spatial registration particularly for objects in motion. We expect that, by not depending on spatially registered CAD data, we remove the risk of rendering poor AR training experiences and even potential safety

issues, which might arise due to imprecise world registration. We rely therefore our AR training proposal on low-cost visual assets, including text, image, video, and predefined auxiliary content.

The last aspect that we considered was the content registration, a core function of most AR systems, still an open issue of research. We identified three main types of information registration methods for HMD-based AR: object, head and environment-based [43]. Marker-based represents the most utilized (57%) registration technique among industrial applications [42]. Other techniques - i.e., 2D/3D recognition, sensor-based, location-based and marker less - do not comply with industrial requirements and are generally limited to test environments [42]. To address robustness and precision requirements, our training proposal relies on head (head-gaze technique) and environment (marker-based technique) registration methods.

2.2. UX Design Principles

User acceptance is identified as one of the most important success factors in the literature [26][27] and during our case study as well. Our informal experiments performed with shop floor workers suggest that a simplistic user experience (UX) is likely the best, considering the profile of the end users and the organization of the manufacturing environment. To ensure the usability of the proposed training method, we adopted a HCD approach: from the usage perspective, the proposed authoring tool should allow shop floor experts easily capture their assembly expertise, independently on the assembly environment. A standalone application, which does not require additional steps (i.e., desktop preparation, fine-tuning or offline content capture) and can be operated in-situ, in a "What You See Is What You Get" (WYSIWYG) manner, is potentially the most adapted. Lee *et al.* [18] demonstrated the advantages of immersive AR authoring in one of the first AR studies of this kind. Recently, Lorenz *et al.* [22] suggested that workstation experts are the most suited to create the AR instructions while better visualization techniques are needed during the authoring process, claims supported by our informal experiments as well.

Further, we analyzed and adopted information-presentation methods (i.e. registration, media types, semi-transparent effect and rotation) proposed as guidelines for AR assembly task support [43] and explored information access and peripheral awareness methods discussed in a study related to information access methods for HMD AR [23]. We followed and adopted guidelines to ensure the usability and effectiveness of the proposed solution, on the shop floor, independently on the assembly environment. We finally designed a hybrid solution, by combining and adjusting these guidelines [43] and techniques [19], to provide a contextualized information conveyance method adapted to the considered manual assembly scenario. We used implicit interaction techniques, including eye tracking and head position, along with common interaction techniques [36] like speech and touch, information outlined in Table 1.

A summary of the main HCD principles around which our proposed AR training system was elaborated, is listed further:

- **Familiarity**: use familiar UI patterns (buttons, arrows) and assets (text, images, and video) to increase user confidence and trust during the usage of the application.
- **Guidance**: use visual cues and implicit interaction techniques to guide the user during the training procedure, in the least intrusive manner.

- **Simplicity**: use a standard information delivery method regardless the variety of the assembly operations. Require deliberate input from the user only when necessary.
- **Comfort and safety**: do not clutter the UI and render the AR content at key locations of the assembly environment, as indicated during the authoring.

2.3. Methodology

The literature already shows that conveying instructions via AR produces better results when compared to classical training procedures [44][7][2]. However, previous research work does not yet provide optimal, standardized ways of delivering step-by-step instructions via AR, even less regarding the authoring of these AR instructions. It is not clear thus which AR visual modalities are optimal for conveying manual assembly information, especially under industrial requirements and challenges. Further, we describe in detail our proposed methodology, which aims to address this research question for the considered boiler-manufacturing use case.

The 2W1H (What, Where and How) Principle In the absence of a standardized method for digitally capturing and conveying manual assembly instructions in AR, we propose a technique that aims to address this concern. We note that each assembly operation, independently of its type and complexity, can be described by three variables: *what*, *where* and *how*. By using this technique, we try to replicate the oral human-to-human explanation of manual operations, as noted during our assembly training experiment and observations. What briefly describes the assembly operation, where indicates the physical location of the assembly operation and finally, how describes how the assembly is performed. This approach is based on the principle proposed by the Greek philosopher Aristotle, known as the "Five Ws (*Who*, *What*, *When*, *Where* and *Why*) and *How*", which represent the six basic questions in problem solving. In the considered use case, *who* – the trainee, *when* – now and *why* – training/authoring procedure, are known, therefore not considered as variables. Our hypothesis is that by following the 2W1H principle, the authors of the AR instructions will be able to describe any manual operation effectively and in a formalized manner, independently on the assembly environment and process. We aim as well to ensure a simple and consistent assembly information conveyance via AR, potentially easy to follow by novice shop floor workers, generally people without technical or AR expertise.

Assembly Instructions Chunking For the 2W1H principle to be applicable, each AR instruction should describe a single assembly operation. As an example, the assembly instruction "*Grab an upright and place it on the structure*" as defined in one of the existing paper instructions analyzed during our case study, becomes two separate "2W1H-friendly" instructions: (1) "*Grab an upright*" and (2) "*Place the upright on the structure*". By using this technique, we expect multiple benefits, as follows. First, the authoring and the training procedures are formalized and consistent, independently on the assembly environment or process. Secondly, by asking the author (during the authoring of the AR instructions) and the trainee (during the training procedure) to perform a single task at a time potentially decreases the assembly complexity, the mental workload, and the error rate. Finally, by limiting the number of virtual elements we avoid the UI clutter. Benefits of a similar chunking technique were recently discussed by Tainaka *et al.* in [43].

Visual Representation of an Assembly Task Regarding the visual representation of assembly tasks, we apply the 2W1H principle for describing them by using the considered low-cost visual assets. Each assembly operation is therefore visually composed of three elements:

- A text instruction, briefly describing the assembly operation (**what**).
- An arrow pointing to the physical location of the assembly operation (**where**).
- A FPV image or video illustrating complex assembly operations (optional) (**how**).

3. System Implementation

To evaluate our proposal, we developed two applications: (i) one for capturing the expert knowledge in AR (see Section 3.1) and (ii) one for conveying the authored AR instructions for training purposes (see Section 3.2). Both applications were developed for Hololens 2 by using Unity 3D (v. 2019.4.10f) [45] and MRTK v. 2.4.0) [32].

3.1. AR Instructions Authoring (Authoring Tool, On-the-Fly, In-Situ - ATOFIS)

The AR device on which we rely the implementation of our proposed authoring method is Hololens 2, which supports text insertion, photo, and video capture, as well as spatial registration of the virtual elements. Such functionalities supported the development of a standalone AR authoring tool that allows shop floor experts to capture their expertise *in-situ*, directly and only inside the AR device. The authoring is a procedure that does not require any prerequisites except an AR device connected to the internet and a unique (per workstation) QR code. The authored AR instructions are ready to be used for training immediately, as soon as the authoring procedure is finalized. Further, let us describe how the content authoring is performed for a single instruction (see Fig. 1), by following the proposed 2W1H principle, as discussed in Section 2.3. The same process applies for creating any AR instruction.

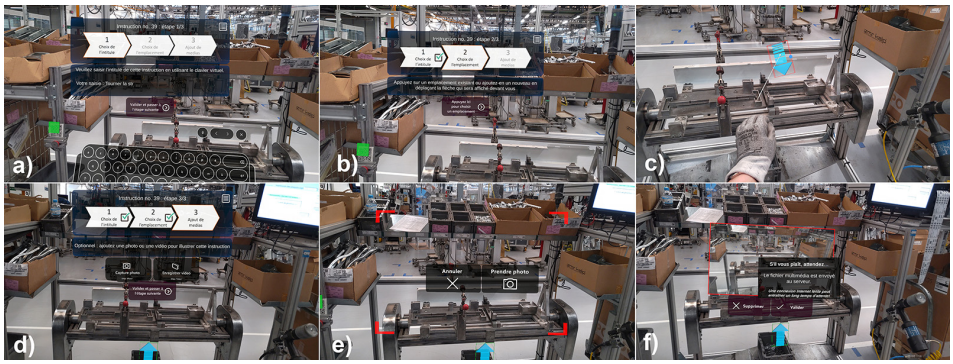


Fig. 1. AR authoring example. a) Step 1. Insert text instruction by using the virtual keyboard or dictation; b) Step 1 validated, step 2 active; c) Step 2. Positioning of the location arrow by using far interaction technique; d) Step 2 validated, step 3 active; e) Step 3. Photo-capture view; f) Step 3. Photo taken, the author validates or removes it.

At any time during the authoring procedure, the author uses hand gestures and voice commands (see Table 1) in order to interact with a 2D panel (Fig. 1.a), b), d)), displayed in front of him by using head registration technique (see Section 2.1). The authoring panel has multiple functions, as further detailed. Firstly, it displays the current assembly instruction number and the authoring step within the current AR instruction. Secondly, it allows the author to access the AR functions for text insertion and FPV photo and video capture. Finally, it allows the author to validate the captured data, advance to the next authoring step and create a new AR instruction. The application implements functions like visualization, selection, and editing of existing AR instructions; however, these are not discussed in the present paper, as they are not essential for the authoring procedure itself. Further, let us describe how the author creates a step-by-step AR instruction by following the 2W1H principle.

What: At this stage (1/3), along with the 2D authoring panel, a virtual keyboard is displayed in front of the user (Fig. 1.a). The author uses the keyboard to insert a text for briefly describing the current assembly task, by using one of the two modalities: (i) natural hand gesture technique, which require touching the virtual keystrokes or (ii) dictation by using voice, a function that is activated by the user by clicking the microphone button, part of the virtual keyboard. The user goes to the next authoring step by clicking a validation button, part of the 2D panel.

Where: At this stage (2/3), the author is required to place a virtual arrow for indicating the physical location of the assembly task (Fig. 1.c)). The arrow is displayed in front of the user, as a static object. The author uses his hand [31] to grab, place, scale and rotate it. Finally, the author validates its and implicitly the authoring step by clicking a "validate" button displayed under the arrow.

How: At this stage (3/3), the author captures a FPV image (Fig. 1.d)) or demonstration video to describe the assembly operation. It is up to the author to decide whether an image, a video or none of the two is required along with the text description and the indication arrow to effectively describe the assembly operation. The author captures one of the two by using the corresponding buttons of the authoring panel or the voice commands: "*photo*", "*video*" and "*stop video*". The author spatially registers the captured media at a convenient location in the real world by using hand interaction techniques [31]. The author is supposed to position the media preferably in the same field of view with the assembly location, to limit the head movement during the training and potentially decrease the assembly time and the effort required by the trainee while following the AR instructions. We note that during training, the images and the videos automatically rotate towards the user; therefore, during authoring, the author should not spend time rotating these elements to face specific real-world locations.

The contextualization of the media elements is one of the main differences between our approach and state-of-the-art AR authoring tools like Guides. Our previous work [15] suggests that the author should decide where the augmentation media is presented during training, instead of teaching and allowing trainees to interact and change its position. Another significant difference compared to existing authoring tools is represented by the way that the AR instructions are created, in a *WYSIWYG* manner, allowing the user to author the AR instructions during the assembly process, to visualize and validate his creation right away, during the authoring itself.

3.2. AR Instructions Conveyance (Training)

Further, let us detail how the assembly information is conveyed and how the user interacts with the visual elements during training. An example of the proposed training interface is presented in Fig. 2. Note that Fig. 2.d illustrates the usage of a CAD model, which replaces a location arrow (see Section 4.2). We note that all instructions are conveyed in the same manner and that the same AR device used for authoring the AR instructions, Hololens 2, is used for conveying these instructions, as follows.

What: Each instruction starts by displaying a text panel (Fig. 2.a)) in front of the user, between 0.6 to 0.7 meters away. The text panel follows user's head for 1 second (head registration) then it stops (environment registration). We ensure that the text is not overlooked by the user and, at the same time, that the panel does not visually interfere for more than necessary. The "sticking time" of 1 second is adjusted for our use case, based on the required movements of the user during the assembly procedure. The user hides the panel by clicking a "hide" button or by using the voice command "hide". Complementing the "hide" button with a voice command was required for cases when the text panel is rendered behind the physical environment, unreachable to hand touch. Our use case validates thus the requirement of multimodal interfaces discussed in [13].

Where: The next step consists in identifying the assembly location, pointed at by a spatially registered arrow (Fig. 2.b)). If the location is not in the field of view (FOV) of the user, a fixed-screen registered arrow (Fig. 2.a) and c)) guides the user towards it. Other techniques for localizing out-of-view objects in AR, like EyeSee360 and audio-tactile stimuli [24] and the "virtual tunnel" [11] are proposed in the literature. However, we rely on the arrow guidance-based technique for several reasons: (i) arrows are familiar visual cues, potentially easy to follow in unfamiliar environments like AR; (ii) visually, arrows are less intrusive and easier to integrate with other AR graphical elements; (iii) the technical implementation of such technique does not represent a challenge. A similar spatial cue technique was recently proposed in [19].

How: Optionally, a FPV image or demonstration video (Fig. 2.d)) describing the assembly operation is displayed in the proximity of the assembly location. Its position is spatially registered by the author so that the visual element and the assembly location are in the FOV of the user, minimizing therefore user's head movement while switching the attention between the two. The eye gaze controls the video playback, meaning that the video plays as long as the user looks at it. The implicit video playback interaction technique allows the trainee to follow video instructions without requiring deliberate input. We address thus the hands-free requirement while avoiding the UI clutter with the classical visual playback controls. We respect the principles discussed in Section 2.2, by allowing the trainee to focus on the assembly process, not on the application usage.

The user visualizes the next/previous instruction by clicking the "next"/"previous" button or by using the corresponding voice command. The "help me" voice command brings the text panel in front of the user. We note that unlike [19], our proposal uses text and images, in addition to video and indication arrows. In our approach, the FPV video is presented during the training experience exactly as captured in the authoring procedure.

Fig. 2 presents an example of the training workflow for performing two assembly operations. The first assembly task requires the worker to grab two uprights from the storage area (Fig. 2.a), b)). The second one requires the worker to place one of the uprights

on the mobile assembly structure of the workstation (Fig. 2.b) and c)). We note that the directional arrow is orange and horizontal while the location one is blue and vertical.

Fig. 2. AR training example: a) Instruction 1a: text description ("*grab 2 uprights*") & directional arrow; b) Instruction 1b: Location arrow & FPV image illustrating the operation; c) Instruction 2a: text description ("*place the 1st upright*") & directional arrow; d) Instruction 2b: Location arrow & FPV video demonstrating the assembly operation.

3.3. Visualization and interaction techniques

Further, let us present the set of visualization and interaction techniques that aims to make it easy for shop floor workers to understand the training interface and be able to follow the AR instructions in the least intrusive manner.

Speech and touch: the user hides the text panel by clicking a "hide" button or by saying the voice command "hide". Similarly, we use voice commands to complement the instruction navigation buttons "next" and "previous". Touch and voice are interaction modalities that complement each other. Our UI requires multimodal interactions for cases when the virtual elements are unreachable by hand. Finally, the "help me" voice command brings the text panel in front of the user if this was hidden or left out of sight.

Head gaze: an implicit interaction technique, used to place dynamically the virtual elements, based on user's position and orientation. We used this technique for placing the text panel in front of the user and for rotating the virtual elements to always face the user. This way, the user is not required to move to certain physical locations for reading the text instruction, watching the associated FPV video demonstration, or inspecting the image.

Eye gaze: another implicit interaction technique that we used to control the video playback involuntarily: the video plays as long as the user looks at it. This technique addresses the hands-free UI requirement identified during our case study. At the same time, this allowed us to remove the "classical" video playback elements, avoiding therefore cluttering the UI. By using implicit interaction techniques, we aim to minimize intentional

input from the user and provide guidance in the least intrusive manner. Table 1 presents a summary of the interaction techniques used during both the authoring (A) and training (T) procedures. Table 2 summarizes the information conveyance workflow. We use the frame of reference (FoR) notation for referring to the registration methods: screen-fixed (SF) and world-fixed (WF) [8].

Table 1. Interaction techniques.

Interaction technique	User input	Ouput
Speech	(A) "*photo*", "*video*", "*stop video*"	FPV image and video
	(T) "*next*", "*previous*", "*hide*", "*help me*"	Instruction navigation, show/hide virtual elements
Touch	Virtual elements (e.g., buttons)	(A) Add/update text instruction, add/replace location arrow, take a photo, record a video (T) Hide visual elements, step navigation
Hands object manipulation	(A) Instinctual interaction [41]	Scale, rotate and move virtual elements
Head gaze	(A+T) Implicit interaction	Dynamic positioning of virtual elements
Eye gaze	(A+T) Implicit interaction	Video playback

Table 2. UI and information conveyance

2W1H	Media type	FoR	Information	User action
What	Text instruction	SF / WF	Briefly describes the assembly operation	Reads text, then hides or ignores the panel
Where	Indication arrow	SF	Guides the user toward the assembly location	Turns the head towards the indicated direction
	Location arrow	WF	Indicates the assembly location	Identifies the location
How	Image / video	WF	Illustrates the assembly	Performs the assembly

4. Field Experiments

We conducted two field experiments and one field study in the boiler-manufacturing factory where we conducted our long-term case study to (i) measure the effectiveness and usability of the proposed AR training method and to (ii) validate our hypothesis that low-cost visual assets are sufficient for describing and conveying complex assembly operations via AR. The first field experiment was a preliminary one, with the main objective of measuring the usability and the effectiveness of the low-cost visual assets for conveying manual assembly instructions. The second experiment, an extension of the first, had as main objectives to (i) assess the potential benefits of authoring CAD-based AR instructions and to (ii) validate the hypothesis that our low-cost-based approach is potentially the most adapted technique for conveying assembly information in similar industrial use cases, until significant AR technical concerns are addressed.

We note that the AR instructions used in all three experiments were created by using ATOFIS, an implementation of the proposed authoring method, discussed in Section 3.1.

However, we discuss authoring statistics exclusively in the study presented in Section 4.3, specifically conducted for this purpose. As the first two field experiments, FE1 (see Section 4.1) and FE2 (see Section 4.2), were conducted under the same assembly setup, we present relevant information regarding the two of them jointly, as follows. Both experiments concerned the assembly of a boiler frame. The procedure consisted of 38 assembly tasks performed on the mobile structure of the first workstation of a manual assembly line. We grouped the assembly tasks (ATx) into four types:

- 14 x AT1 – picking (assembly components and tools)
- 8 x AT2 – installing / placement (assembly components)
- 12 x AT3 – screwing & riveting (screws and rivets)
- 4 x AT4 – manipulating (assembly structure and tools)

We used ATOFIS to author two sets of AR instructions. The first set, evaluated in FE1, was based solely on low-cost visual assets. The second instruction set was identical with the first, except that CAD models replaced the location arrows in assembly instructions of type AT2. The field experiment FE2 evaluated and compared both instruction sets. We note that every AT2 instruction had a FPV demonstration video associated to it in the first instruction set, complemented with a CAD model in the second. There was no potential benefit in complementing with CAD models the other assembly types (AT1, AT3 and AT4), reason for which these instructions were the same between the two instruction sets.

4.1. Field Experiment 1 (FE1)

The main objective of the field experiment presented in this section was to evaluate the effectiveness of the proposed low-cost-based AR training approach. A comprehensive description of the experiment set-up, participants, evaluation procedure, results and conclusions is presented in our previous work [16]. In this section, we present a summary of the most relevant findings reported in field experiment.

In the absence of an agreed upon framework to assess AR training systems for manual assembly process, we adopted the two evaluation methods identified by Wang *et al.* [47] in their AR assembly research survey: effectiveness and usability. We assessed the effectiveness of the proposed training method by measuring the error rate, the assembly completion time (ACT) and the instruction reading time (IRT). ACT represents the time spent for completing an assembly task; IRT represents the time spent on reading the low-cost visual assets. We evaluated the usability of the proposed training method by using the System Usability Scale (SUS) questionnaire [3]. We note that in FE2 (see Section 4.2) the extension of FE1, we present the evaluation results that include those reported in FE1, reason for which this section does not present in detail the findings further discussed.

Our measurement reported that 75% (9 out of 12) of the participants committed one or two errors during the first assembly cycle; however, we observe that the average error rate of 2.63% is very low, a value which potentially suggests a lack of user attentiveness rather than an issue regarding the AR information conveyance method. The convergence of the error rate to zero and the progress of the ACT over the course of the three assembly cycles potentially demonstrate the usability and effectiveness of our proposed training method. We note that all the reported errors except one consisted in a wrong orientation

of the assembly component, supporting thus the effectiveness of the proposed AR training method, particularly for tasks of type AT1, AT2 and AT4.

A relevant finding revealed during the experiment indicates that participants get familiarized and recall the assembly instructions at a very fast pace: the time spent on reading the AR instructions decreases by 60% in the third assembly cycle, indicating a rapid diminishing utility of the AR instructions. The worthiness of authoring AR instructions by using CAD data, animations and other "expensive" media that would make the authoring more laborious is therefore questioned, a claim that is partially demonstrated in the second field experiment, FE2, discussed in the next section.

Fig. 3 illustrates the time required by each participant for completing each assembly task over three assembly cycles. We note the average time spent by the participants to read the instructions, *AvgRead* (blue line), matching closely the video length, *Video* (black line), during the first assembly cycle. We observe as well that *AvgRead* is flattening over the course of the three cycles. The peaks of *Video* over *AvgRead* in Fig. 3.c) indicate that participants stop watching the video entirely at this point, potentially suggesting that after only two cycles, videos can be replaced by images, even removed. The flattening of *AvgTotal* (red line), indicates the learning progress and the familiarization with the assembly.

Finally, we used the SUS questionnaire (see Table 7) to evaluate the overall usability of the proposed training method, by using a five-item Likert scale ranging from "*strongly agree*" to "*strongly disagree*". The reported usability score was 88.33 (SD = 9.02). All the participants that ranked Q4 with a score <= 60, claimed that they might need human support during the first assembly cycle, further referred to as the workstation exploration cycle (WEC). This claim is supported by the reported error rate during the WEC, where 75% of the participants committed one or two errors

The findings of the field experiment presented in this section suggest that spatially registered 2D visual assets together with a specific, human-centered set of interaction and visualization techniques could provide an effective AR-based conveyance method for describing manual assembly operations in industrial context. We note however that the reported error rate during the WEC suggests that a better technique for describing error-prone assembly operations is required, particularly for novice operators and during WEC.

4.2. Field Experiment 2 (FE2)

The field experiment presented in this section had multiple objectives, further listed:

- (O1) Extend and demonstrate the findings of FE2.
- (O2) Identify potential benefits of using CAD models for manual conveying assembly information by comparing the usability and effectiveness of two sets of AR instructions: low-cost vs. CAD-complemented.
- (O3) Answer a research question [9] suggesting that studies are needed to identify optimal ways to convey instructions in industrial sectors via AR.
- (O4) Validate the HCD principles discussed in Section 2.2, by measuring the perceived usability of the system and the mental workload of the participants.

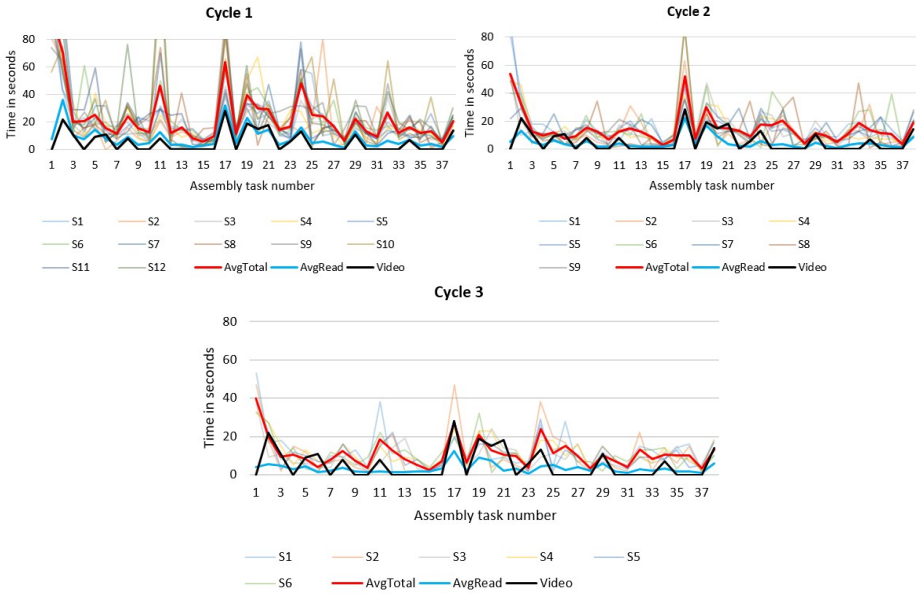


Fig. 3. Average assembly (*AvgTotal*) and reading (*AvgRead*) times per instruction, per a) cycle 1, b) cycle 2 and c) cycle 3

Similarly, as for the FE1, in this paper we present the most significant aspects of the field experiment FE2. A comprehensive description of this study is presented in our previous work [17]. We note that FE2 extends FE1 from 12 participants to 30 and from one instruction set (low-cost-based) to two instruction sets (low-cost and CAD-based). We created two groups, **G-LA** and **G-CAD**, each composed of 15 participants, for evaluating the two instruction sets: **LA** = Low-cost Assistive-based instruction set and **CAD** = CAD-based instruction set. Five participants have assembly experience in each group. We created two subgroups for each group: **G-LA-N** = novice participants from G-LA and **G-LA-E** = experienced participants from G-LA. Similarly, for G-CAD: **G-CAD-N** and **G-CAD-E**. We grouped the participants as such, to identify if assembly experience has a notable influence on the training performance.

Table 3 outlines this information.

Table 3. Evaluation groups

Group	**G-LA**		**G-CAD**	
Subgroup	G-LA-N	G-LA-E	G-CAD-N	G-CAD-E
Number of participants	10	5	10	5
Assembly experience	No	Yes	No	Yes
Instruction set number	1		2	

We assess the effectiveness and the usability of the proposed training system in the same manner as described in FE1, per instruction set and per subgroup, to address the

main four objectives aforementioned. Further, we present a summary of the most relevant findings of this field experiment.

Table 4 presents the number of participants, per group, performing the nth assembly cycle. For each assembly cycle, we present the percentage of participants committing errors, the average error rate per instruction set, the total ACT and IRT (% of the ACT), and finally the average ACT of assembly operations of type AT2. We measure the IRT to identify differences and to estimate the utility of low-cost visual assets over multiple assembly cycles, within each instruction sets. A comparison between the two instruction subsets of type AT2 (CAD-complemented) is performed separately. Table 4 however presents the reported data collected on all instructions, to identify the impact of the CAD-based instructions over the whole instruction set, a practical evaluation approach for the considered use case.

The reported error rate and type shows that except one, all assembly errors were committed on operations of type AT2, during the first two cycles. We observe that subtle assembly details are prone to be overlooked, especially by participants without assembly experience, which commit more errors, as shown in Table 5, reason for which we believe that a better visual modality is needed for highlighting key assembly details, particularly for novice workers, during the WEC.

Table 4. Evaluation measurements

Group	**G-LA**			**G-CAD**		
Cycle no.	1	2	3	1	2	3
Participants no.	15	12	8	15	15	9
Participant error rate (%)	66%	25%	0%	66%	20%	0%
Total number of errors	13	3	0	18	3	0
Error rate per set (%)	2.2%	0.6%	0%	3.1%	0.5%	0%
Avg. ACT (s)	884s	538s	367s	838s	475s	336s
ACT progress (nth-1)		39%	31%		43%	29%
Avg. IRT (%)	37%	29%	25%	31%	27%	19%
Avg. ACT of AT2 (s)	290s	165s	98s	268s	130s	74s

Table 5 presents the average error rate committed per participant in each subgroup during the WEC. The error rates of the following cycles are not significant, therefore not discussed.

Table 5. Error rates per subgroup during the WEC

Group	**G-LA**		**G-CAD**	
Subgroup	G-LA-N	G-LA-E	G-CAD-N	G-CAD-E
Avg. errors per participant	1.1	0.4	1.4	0.8
Novice vs. experienced	-63%		-42%	
G-LA vs G-CAD	+38%			

We note that participants with assembly experience commit fewer errors in both groups (-63% and -42%) and surprisingly that G-CAD commits more errors than G-LA (38%).

The IRT measurement reveals that G-CAD participants watch the instructional videos less (during AT2 instructions) that G-LA, relying therefore on the CAD information more, potentially explaining their higher error rate during the WEC. The reported measurements indicate that G-CAD participants use 29% less time for watching videos, leading to a 7% decrease in the ACT, but to an increase in the error rate by 38% (see Table 5). The error rate convergence to zero after three assembly cycles support the hypothesis that both instruction sets are reliable for conveying manual assembly information in the considered use case. However, for assembly tasks of type AT2, the evaluation results suggest that human supervision might be necessary during the WEC.

The mean (M) and the standard deviation (SD) between the ACT over the three assembly cycles presented in Table 6, support the WEC paradigm and indicate that the participants start familiarizing with the assembly operations at a rapid pace. These findings underpin the conclusion of the FE1 and support the hypothesis that questions the worthiness of authoring of CAD-based AR instructions in similar industrial use cases.

Table 6. Mean and standard deviation of the ACT and IRT over three assembly cycles

Cycle number	**1**		**2**		**3**	
Global ACT/IRT	ACT	IRT	ACT	IRT	ACT	IRT
M	22.67	7.85	13.34	3.80	9.26	2.07
SD	13.04	6.89	6.73	3.07	4.04	1.32

Fig. 4 illustrates the mean ACT of all participants, per cycle. The ACT "flattening" over the three assembly cycles supports our claim, indicating the learning progress.

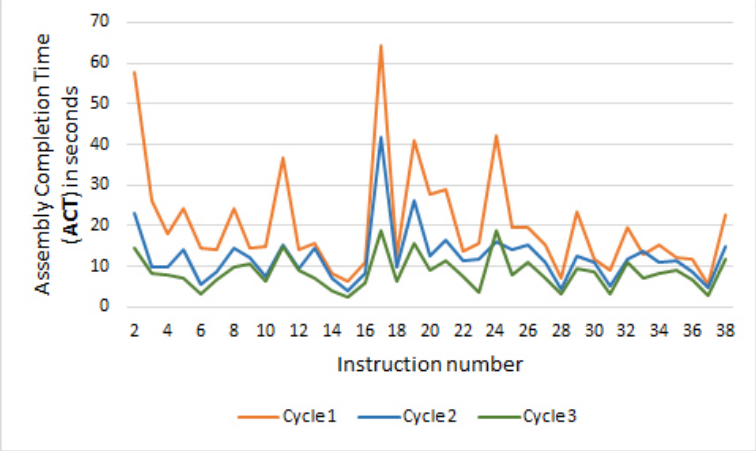


Fig. 4. ACT per instruction over 3 assembly cycles

The overall value of the IRT decrease by 47.5% and by 35.7% in the 2nd and 3rd cycles indicates that participants become less dependent on the AR instructions rapidly (see Table 6). By considering only the AT2 subset, we observe that CAD-based instructions

lead to a faster assembly progress: G-CAD requires less time to perform the assembly operations of type AT2 compared to G-LA: -7%, -20% and -22% over the three assembly cycles (see Table 4).

In addition, we used the Pearson's correlation test to analyze the correlation between the ACT performances between the two groups, over the assembly operations of type AT2. The test shows a high correlation between the mean ACT of the two groups, during all assembly cycles: [r = 0.94, p = 0.142] for the first cycle, [r = 0.99, p = 0.003] for the second one and finally [r = 0.96, p = 0.007] for the third assembly cycle. We observe a high correlation (r = 0.94) without statistical significance (p = 0.142) during the first cycle; however, very strong correlations with statistical significance are reported for the second and third assembly cycles, respectively.

Finally, a subjective evaluation of the training method, including both instruction sets was performed. We used the SUS questionnaire (see Table 7) to evaluate the overall usability of the proposed training method (see Fig. 5).

Table 7. SUS questionnaire used to evaluate our proposed AR training system

No.	Question
1	I think that I would like to use this system frequently.
2	I found the system unnecessarily complex.
3	I thought the system was easy to use.
4	I think that I would need the support of a technical person to be able to use this system.
5	I found the various functions in this system were well integrated.
6	I thought there was too much inconsistency in this system.
7	I would imagine that most people would learn to use this system very quickly.
8	I found the system very cumbersome to use.
9	I felt very confident using the system.
10	I needed to learn a lot of things before I could get going with this system.

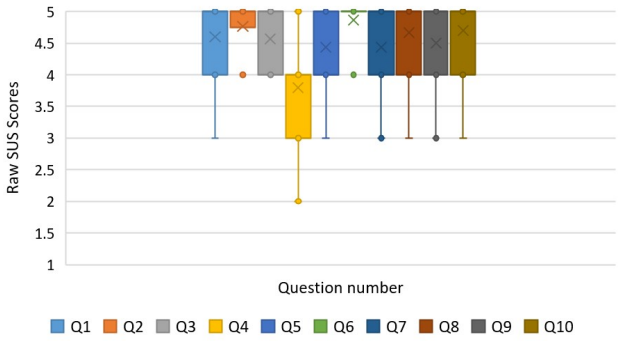


Fig. 5. Reported overall raw SUS scores

A one-way analysis of variance reveals no significant differences between G-LA and G-CAD [F(1,28)=0.01, p=0.89)] or between G-Experienced and G-Novice [F(1,28)=0.71, p=0.40)]. The overall reported perceived usability for all the participants is 4.53 (SD=0.25),

indicating that the proposed method validates the HCD principles presented in Section 2.2. Similarly, as in FE1, Q4 reports the lowest rating: S=3.80, underlining the claim that human supervision is required during the WEC. However, Q4 reports a significant difference between G-LA vs. G-CAD [F(1,28)=5.34, p=0.02)] potentially indicating that CAD-based AR instructions lead to higher user confidence, evidence supported as well by the IRT difference of assembly operations of type AT2.

Further, we used the NASA-TLX questionnaire [12] to measure the mental workload of the participants. The raw NASA-TLX scores reported the following values: S=24.42, SD=4.75 for G-LA; S=24.22, SD=5.00 for G-CAD; S=25.25, SD=6.13 for G-Experienced and S=23.85, SD=5.49 for G-Novice (see Fig. 6). A one-way analysis of variance (ANOVA) finds no statistically significant differences ($p>.05$) between all groups and on all dimensions. However, our post-experiment evaluation reveals that participants with assembly experience have higher expectations from a temporal perspective, affecting their perceived performance level.

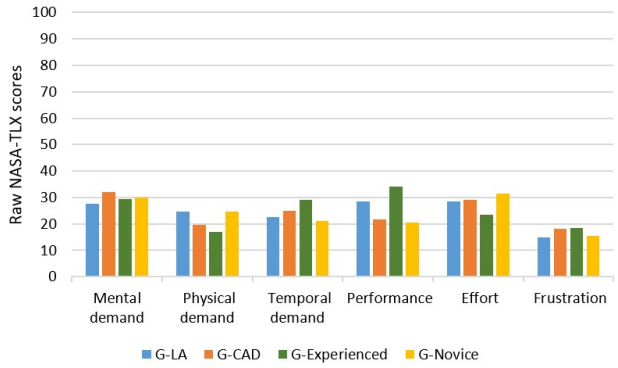


Fig. 6. Raw NASA-TLX scores per dimension and per group

4.3. Field Study (FS)

The study presented in this section was conducted by one of the authors and aimed at evaluating the proposed AR authoring system, ATOFIS (see Section 3.1), particularly from a time perspective. We conducted this work to provide an estimate of the time required for authoring the AR instructions of an assembly workstation and, concurrently, to compare our proposal with one of the most representative industrial AR authoring systems. We analyzed the creation of step-by-step low-cost-based AR instructions by measuring their authoring time and their media type composition. We note that the current study does not deal with CAD data, nor does it evaluate the authoring difficulty. Unlike FE1 and FE2, this section discusses unpublished data; therefore, a complete description of the considered study is presented, as further described.

Study set-up and evaluation procedure

The study was conducted in the same industrial environment as FE2. Five workstations

have been uniformly chosen for authoring the corresponding step-by-step assembly instructions in AR. The numbers of assembly operations per each workstation are 33, 11, 33, 20 and 35. The author of the instructions is expert in performing the assembly tasks of the selected workstations and in manipulating the AR authoring system. We expect therefore the collected data of the field study to provide a reliable assumption regarding the authoring procedure when performed by experienced shop floor workers (e.g., line manager), at ease with the proposed system. In addition, this study aimed to validate authoring expectations from an industrial perspective, particularly regarding time constraints, adaptability, and robustness, before conducting a large-scale field experiment with shop floor experts, a costly and difficult study.

To evaluate objectively our proposed authoring method from a time perspective, we decided to compare it against Guides, the most popular state-of-the-art AR authoring tool. We do not detail the authoring workflow of Guides, available at [34]; however, we note the authoring workflow of Guides being comprised of three stages: media capture, PC authoring and HMD authoring. To avoid bias, for each workstation we wrote down the assembly instructions, together with the media type that shall be captured and the number of location arrows. This way we aimed to ensure the creation of the same AR instructions with both authoring systems: text description, number of location arrows and image or video (see the 2W1H principle, Section 2.3). Any assembly instruction was therefore written down by respecting the following template: "*DESCRIPTION, M, N*", where "M" can be either I (image) or V (video) and represents the captured media type while N is an integer number and represents the number of the location arrows of the assembly task in question. An example of such an assembly instruction is "*Screw the 4 screws at the indicated locations, I, 4*". To avoid bias, for every other workstation, the authoring started with the last system used for authoring the previous one.

Results and Interpretation

Table 8 presents the number of AR instructions and their corresponding media assets and characters, per workstation. Together with the information displayed in Table 9, we estimate what an AR assembly instruction is composed of, on average. Further, by considering the information presented in Table 11, we estimate the time required for authoring the AR instructions for a new workstation.

Table 8. Authoring data captured per workstation

Workstation	**W1**	**W2**	**W3**	**W4**	**W5**	**Total**
No. of instructions	33	11	33	20	35	132
No. of videos	12	6	13	9	9	49
No. of images	17	5	16	11	19	68
No. of arrows	43	22	41	24	45	175
No. of characters	1214	358	794	347	1106	3819

Table 9 presents the average number of characters, location arrows, images and videos used to author an AR instruction, per workstation.

Table 9. Average number of media used for authoring an AR instruction, per workstation

Workstation	W1	W2	W3	W4	W5	Total avg.
Avg. per instruction	33	11	33	20	35	132
No. of characters	37	33	24	18	32	28.8
No. of location arrows	1.3	2	1.2	1.3	1.3	1.4
No. of images	0.51	0.45	0.48	0.55	0.54	0.5
No. of videos	0.36	0.54	0.39	0.45	0.26	0.4

We used the reported preliminary data to estimate the composition, on average, of an AR instruction and the overall required authoring time of a new assembly workstation. The average number of videos per workstation indicates that the author of the AR instructions considers that 40% of the total assembly tasks are difficult or error-prone and require a video demonstration. The average number of images indicates that 50% of the total assembly tasks are relatively easy and that 10% of them are obvious and do not require a visual representation. Finally, the number of characters used to describe an assembly task is 28.8, on average, as indicated in Table 9.

Table 10 presents the authoring times in seconds, per workstation, for both AR authoring systems, Guides and ATOFIS. Additionally, the time differences, in percentage, in the favor of ATOFIS are presented.

Table 10. Authoring times comparison between ATOFIS and Guides

Workstation	W1	W2	W3	W4	W5	Total
Guides (seconds)	2822s	991s	2459s	1804s	2706s	10782s
ATOFIS (seconds)	1805s	606s	1663s	1058s	1837s	6969s
Difference (%)	-36%	-39%	-32%	-41%	-32%	-35%
No. of images	0.51	0.45	0.48	0.55	0.54	0.5
No. of videos	0.36	0.54	0.39	0.45	0.26	0.4

We observe an authoring time improvement of 35% on average, in the favor of ATOFIS. We speculate therefore that, from a time perspective, our proposed authoring system will outperform Guides by 35% on average in a similar context. By considering the low number of data points (5), a definitive claim cannot be made, however.

Further, a remarkable finding observed from the time measurements (see Table 11), where the average authoring times required to create a single AR instruction, per workstation, by using ATOFIS and Guides, respectively, are presented. It is interesting to note the extremely low standard deviation (SD =1.32) and variance (VAR=1.76) between the average authoring times of an AR instruction, for all workstations, reported by ATOFIS.

By considering the fact that the evaluated workstations are representative, they were uniformly selected, and their cycle times is very similar (guaranteed by the assembly line balancing procedure), we could expect that the authoring time of a new workstation, Wx, composed of N assembly instructions, will approximately be N*53.2 seconds. We highlight however, the fact that our estimation is based on a limited number of data points, five. Unlike ATOFIS, Guides reported higher values for both the standard deviation (SD=6.61)

Table 11. Average AR instruction authoring time, per workstation

Workstation	W1	W2	W3	W4	W5	Total avg.
ATOFIS (seconds)	54s	55s	51s	53s	53s	53.2s
Guides (seconds)	85s	90s	74s	90s	77s	83.2s

and the variance (VAR =43.76). This data suggests that a better authoring time approximation could be estimated for our proposed authoring system, information that plays an important role in the planning and potentially adoption of the proposed AR system in a similar industrial use case.

5. Discussion

5.1. Training Field Experiments (FE1 & FE2)

The percentage of participants committing errors during the WEC, 66% in each of the two groups (L-GA and L-CAD), invalidates the hypothesis that novice workers can perform the training completely unsupervised. CAD-complemented AR instructions lead to faster ACT (-7% in the first assembly cycle, -20% in the second and -22% in the third), but to a higher error rate during the first assembly cycle (+38%). These results suggest that (i) FPV video demonstrations are more reliable for conveying error-prone assembly operations to novice workers and secondly, that (ii) CAD-based instructions lead to faster assembly for operations of type AT2, especially after the WEC. It seems therefore that FPV video demonstrations are more effective for conveying assembly information during the WEC, however leading to higher assembly times in the following cycles, when compared to CAD models. The reported IRT shows that G-CAD uses 29% less time for reading the instructions during the WEC, indicating that participants prefer CAD-based guidance to video demonstrations. By considering the higher error rate and the lower IRT of G-CAD during the WEC, we speculate that CAD models persuade higher user confidence and lower user attentiveness. The ACT decrease of only 7% in the favor of G-CAD and the error rate increase of 38% reported during the first assembly cycle support the hypothesis than video demonstrations are more effective than CAD models for conveying assembly instructions of type AT2 to novice workers. The overall decrease of the IRT by 47% and 35% in the second and third assembly cycles, respectively, suggest that participants become less dependent on the AR instructions rapidly, questioning therefore the worthiness of authoring CAD-based AR instructions in similar industrial use cases.

A comparative evaluation between the authoring of the two instruction sets used in the field experiments FE1 and FE2 was not conducted. However, we strongly argue, based on the work carried out for authoring the two instruction sets, that the overall authoring effort (technical expertise and time) for creating CAD-based AR instructions is significantly higher compared to creating low-cost-based AR instructions. By considering that only a single error of type non-AT2 was reported during all assembly cycles, we anticipate that low-cost visual assets can effectively convey assembly information of type AT1, AT3 and AT4 in similar assembly use cases, even during the WEC. Additionally, all participants,

independently on their assembly performance reported during the experiment, believe that spatially registered low-cost visual assets are sufficient for conveying assembly instructions via AR. Our picking technique differs from the one proposed in [11], however, the effectiveness of pick-by-AR technique is supported by the results of our field experiments.

Assembly experience leads to a better training performance. We observed during the experiment that participants with assembly experience perform significantly faster in AT3 tasks (e.g. screwing or riveting) and commit fewer errors (-52%) during the WEC. They do not commit errors during the following cycles and their ACT is better, independently on the group: -9%, -19% and -16% on average over the three assembly cycles. AR experience however does not affect the performance. We do not observe a lower mental workload nor usability advantages for participants with AR experience, potentially demonstrating the usability of the proposed AR training method.

The ACT of G-CAD suggests that registered CAD models allow a faster identification of the position and orientation of the assembly components, as long as a precise spatial registration is guaranteed. However, the reliability of spatially registered CAD-based AR instructions is questioned until an accurate continuous object registration technique will be provided. We note that the CAD-based AR training experience is highly dependent on the quality of the spatial registration, a concern that was partially addressed in the experiment by the assembly environment itself, as most workstation components had a fixed position, unchanged between the authoring and training procedures. We note as well that CAD models seem to interfere visually with the assembly location, making it difficult to perform operations in non-obvious locations, as reported by few participants. A similar concern was observed in video-based instructions as well, where some of the participants spent more time than expected to identify the corresponding real world assembly location indicated in the demonstration video.

Finally, it seems that successive and repetitive assembly operations like screwing and riveting can be grouped and conveyed as a single AR instruction. Participants performing at least three assembly cycles either suggested or agreed on this affirmation, indicating that the instruction chunking technique was not adapted for certain repetitive assembly operations, particularity after the WEC.

5.2. Authoring Field Study (FS) Experiment

The preliminary data collected during the authoring field study indicates that an expert in manipulating the proposed authoring system, would require approximately 30 minutes for capturing his assembly expertise of a workstation with a nominal cycle time of approximately 3 minutes and 30 seconds (30-35 assembly tasks), in a similar assembly context. The media types captured during the authoring of the five selected workstations for the evaluation indicate that 40% of the assembly operations require video demonstrations, 50% require an image while 10% only require a text description and an indication to the physical location of the assembly. Together with the average reported authoring time per instruction (see Table 11), this information potentially allows one to estimate the authoring time of a workstation that is not representative for the considered use case (e.g., automotive). To do so, the corresponding assembly tasks should be grouped by difficulty into: "*require video demonstration*" (complex assembly task), "*require image*" (rather easy task), "*no visual information required*" (elementary/routine task). We are not aware

of a standard method for objectively classifying the assembly operations by their difficulty. Consequently, we believe that the author of the AR instructions, a shop floor expert, potentially an assembly instructor, should decide what type of description each assembly requires for a novice worker to be able to perform it correctly and efficiently.

Secondly, we demonstrated that our proposed authoring method performs, on average, 35% faster than Guides. The preliminary collected data suggests that a precise authoring time estimation of a manual assembly workstation, similar to our use case, can be made by using our proposed method. In addition to the authoring time gain, we expect that our proposal has other advantages including faster learning curve, lower mental and physical effort, less expertise required and ultimately better adapted to industrial usage. In future work we will conduct a participant-based evaluation of the proposed AR authoring system, to validate the aforementioned claims and to measure the usability and the user preference of the system as well.

6. Conclusions

In this research work, we presented an AR training system for manual assembly, adapted to industrial context. We discussed the design and the implementation of the proposed AR authoring tool, dedicated to shop floor experts for capturing assembly knowledge in a one-step authoring process, entirely performed in an HMD AR device (i.e., Hololens 2). Further, we presented how the captured information, represented by a set of step-by-step instructions, is conveyed, and consumed by novice workers via AR, for training purposes. During our long-term case study, we found that, to address industrial challenges and requirements, the best compromise was to rely the proposed AR training system on low-cost visual assets like text, image, video, and predefined auxiliary content, instead of CAD data and animations. To validate our hypotheses, we conducted two field experiments in a real-world industrial use case.

The findings of the first field experiment (FE1) suggested that spatially registered 2D low-cost visual assets are sufficient and effective for conveying manufacturing expertise to novice workers via AR. In the second field experiment, FE2 (an extension of the first), we comparatively evaluated a CAD-complemented instruction set with the initial one (low-cost-based) to identify potential benefits of conveying assembly information by using non-animated, registered CAD models. We found that CAD data persuades lower user attentiveness, eventually leading to a higher error rate for components with a high degree of symmetry (error-prone), but to faster overall assembly completion times, particularly after the WEC. By considering the progress of the time spent by the participants in reading the AR instructions over three assembly cycles, we concluded that the worthiness of authoring CAD-based instructions in similar industrial use cases is questionable, until significant technical and organizational AR challenges are addressed. The overall reported effectiveness and usability scores are favorable, indicating that the proposed AR training method can potentially be used in concrete real world industrial use cases, with a remark that a better technique for underlining subtle assembly details is required for ensuring error-free, completely unsupervised AR training procedures. We expect that our approach can be generalized and adopted in other manufacturing use cases where the 2W1H principle can be applied.

In the third experiment, a preliminary field study (FS), we aimed at evaluating the proposed authoring method from a time perspective, to estimate the required authoring time of a lambda assembly workstation. In addition, we compared our proposal with the most representative industrial AR authoring tool, Guides, and found that our system is 35% faster. Secondly, the evaluation results suggest that our authoring system provides a better time estimation for creating AR instructions, presumably for workstations that are not representative to the considered use case.

Finally, we believe that the industry does not need to wait for better registration techniques, 3D content authoring processes or interfaces. We demonstrated that easy to author, low-cost visual assets together with specific interaction and visualization techniques available in state-of-the-art AR devices could provide effective AR training experiences in complex, real-world industrial environments. At the same time, we demonstrated that organizational and technical AR challenges could be overcome, as long as the conception of the solution is elaborated and tested in the right context, with the direct involvement of the potential end users.

7. Future Work

The main limitation of our work is that the proposed training method was evaluated on a single assembly workstation. To obtain unquestioning statistical data regarding its effectiveness and usability, full training procedures involving novice assembly workers and multiple workstations might be required. We anticipate that future evaluations considering detailed user profiles including cognitive skills, will reveal important findings regarding optimal ways of conveying profile-adapted instructions in AR. From a training perspective, we plan to conduct a large-scale evaluation in other manual assembly use cases and to extend the current evaluation (FE2) to multiple workstations, ideally performing complete training procedures on multiple novice workers.

Regarding the authoring of the AR instructions, in our future work we will aim to evaluate objectively the proposed AR authoring method by conducting a field experiment, ideally by involving line managers and other shop floor experts. Similarity with the first two experiments described in this paper (FE1 and FE2), we will aim to measure the effectiveness and the usability of the overall training system as follows: prior to authoring the AR instructions, participant N will be trained by using the AR instructions created by participant N-1. We will objectively evaluate our proposed AR system against Guides, by alternating the two systems, for both the training and the authoring procedures, for every next participant.

Finally, a comparative evaluation between authoring low-cost versus CAD-based AR instructions might be conducted to objectively evaluate the authoring effort of the two instruction sets and demonstrate our hypothesis, which questions the worthiness of creating CAD-based AR instructions for conveying assembly expertise in similar industrial context.

References

1. Bellalouna, F.: Industrial use cases for augmented reality application. In: 2020 11th IEEE International Conference on Cognitive Infocommunications (CogInfoCom). pp. 10–18 (2020), https://doi.org/10.1109/CogInfoCom50765.2020.9237882

2. Bosch, T., Könemann, R., Cock, H., Rhijn, G.: The effects of projected versus display instructions on productivity, quality and workload in a simulated assembly task. In: Proceedings of the 10th International Conference on PErvasive Technologies Related to Assistive Environments (PETRA '17. p. 412–415. Association for Computing Machinery, New York, NY, USA (2017), https://doi.org/10.1145/3056540.3076189, dOI:
3. Brooke, J.: System Usability Scale (SUS). https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html (1986), online; accessed 2021-07-06
4. Caudell, T., Mizell, D.: Augmented reality: an application of heads-up display technology to manual manufacturing processes (2003), https://doi.org/10.1109/hicss.1992.183317
5. Dey, A., Billinghurst, M., Lindeman, R., Swan, J.: A systematic review of 10 years of augmented reality usability studies: 2005 to 2014. Front. Robot. AI 5 (2018), https://doi.org/10.3389/frobt.2018.00037
6. Egger, J., Masood, T.: Augmented reality in support of intelligent manufacturing – a systematic literature review. Comput. Ind. Eng 140, 106195, (2020-02), https://doi.org/10.1016/j.cie.2019.106195
7. Funk, M., Kosch, T., Schmidt, A.: Interactive worker assistance: Comparing the effects of in-situ projection, head-mounted displays, tablet, and paper instructions. UbiComp 2016 - Proc. 2016 ACM Int. Jt. Conf. Pervasive Ubiquitous Comput pp. 934–939, (2018-07), https://doi.org/10.1145/2971648.2971706
8. Gabbard, J., Fitch, G., Kim, H.: Behind the glass: Driver challenges and opportunities for ar automotive applications. Proc. IEEE 102(2), 124–136, (2014), https://doi.org/10.1109/JPROC.2013.2294642
9. Gattullo, M., Evangelista, A., Uva, A., Fiorentino, M., Gabbard, J.: What, how, and why are visual assets used in industrial augmented reality? a systematic review and classification in maintenance, assembly, and training (from 1997 to 2019. IEEE Trans. Vis. Comput. Graph 2626(c), 1–1, (2020), https://doi.org/10.1109/tvcg.2020.3014614
10. Hahn, J., Ludwig, B., Wolff, C.: Augmented reality-based training of the pcb assembly process. In: Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia (MUM '15. p. 395–399. Association for Computing Machinery, New York, NY, USA (2015), https://doi.org/10.1145/2836041.2841215, dOI:
11. Hanson, R., Falkenström, W., Miettinen, M.: Augmented reality as a means of conveying picking information in kit preparation for mixed-model assembly. Comput. Ind. Eng 113(August), 570–575, (2017-11), https://doi.org/10.1016/j.cie.2017.09.048
12. Hart, S.G.: Nasa-Task Load Index (NASA-TLX); 20 Years Later. In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting. p. 904–908 (2016), https://doi.org/10.1177/154193120605000909.K
13. Irawati, S., Green, S., Billinghurst, M., Duenser, A., Ko, H.: An evaluation of an augmented reality multimodal interface using speech and paddle gestures. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics (January), 272–283, (2006), https://doi.org/10.1007/11941354_28
14. Kim, K., Billinghurst, M., Bruder, G., Duh, H., Welch, G.: Revisiting trends in augmented reality research: A review of the 2nd decade of ismar (2008-2017. IEEE Trans. Vis. Comput. Graph 24(11), 2947–2962, (2018), https://doi.org/10.1109/TVCG.2018.2868591
15. Lavric, T., Bricard, E., Preda, M., Zaharia, T.: An AR Work Instructions Authoring Tool for Human-Operated Industrial Assembly Lines. In: 2020 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR. pp. 174–183, (2020), https://doi.org/10.1109/AIVR50618.2020.00037
16. Lavric, T., Bricard, E., Preda, M., Zaharia, T.: Exploring low-cost visual assets for conveying assembly instructions in ar. In: 2021 International Conference on INnovations in Intelli-

gent SysTems and Applications (INISTA. pp. 1–6, (2021), https://doi.org/10.1109/INISTA52262.2021.9548570
17. Lavric, T., Bricard, E., Preda, M., Zaharia, T.: An industry-adapted ar training method for manual assembly operations. In: HCI International 2021 - Late Breaking Papers: Multimodality, eXtended Reality, and Artificial Intelligence. pp. 282–304. Springer International Publishing, Cham (2021), https://doi.org/10.1007/978-3-030-90963-5_22
18. Lee, G., Nelles, C., Billinghurst, M., Kim, G.: Immersive Authoring of Tangible Augmented Reality Applications Introduction Application Domain Analysis Immersive Authoring Design. Ismar (2004), https://doi.org/10.1109/ISMAR.2004.34
19. Lee, G.A., Ahn, S., Hoff, W., Billinghurst, M.: Enhancing first-person view task instruction videos with augmented reality cues. In: 2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR). pp. 498–508 (2020), https://doi.org/10.1109/ISMAR50242.2020.00078
20. Li, W., Wang, J., Jiao, S., Wang, M., Li, S.: Research on the visual elements of augmented reality assembly processes. Virtual Real. Intell. Hardw 1(6), 622–634, (2019-12), https://doi.org/10.1016/j.vrih.2019.09.006
21. Lopik, K., Sinclair, M., Sharpe, R., Conway, P., West, A.: Developing augmented reality capabilities for industry 4.0 small enterprises: Lessons learnt from a content authoring case study. Comput. Ind 117, 103208, (2020-05), https://doi.org/10.1016/j.compind.2020.103208
22. Lorenz, M., Knopp, S., Klimant, P.: Industrial augmented reality: Requirements for an augmented reality maintenance worker support system. In: Adjun. Proc. - 2018 IEEE Int. Symp. Mix. Augment. Reality, ISMAR-Adjunct 2018. pp. 151–153, (2018), https://doi.org/10.1109/ISMAR-Adjunct.2018.00055
23. Lu, F., Davari, S., Lisle, L., Li, Y., Bowman, D.: Glanceable ar: Evaluating information access methods for head-worn augmented reality. In: 2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), Mar. pp. 930–939, (2020), https://doi.org/10.1109/VR46266.2020.00113
24. Marquardt, A., Trepkowski, C., Eibich, T., Maiero, J., Kruijff, E., Schoning, J.: Comparing non-visual and visual guidance methods for narrow field of view augmented reality displays. IEEE Trans. Vis. Comput. Graph pp. 1–1, (2020), https://doi.org/10.1109/tvcg.2020.3023605
25. Martinetti, A., Marques, H., Singh, S., Dongen, L.: Reflections on the limited pervasiveness of augmented reality in industrial sectors. Applied Sciences 9. 3382 (2019), https://doi.org/10.3390/app9163382
26. Masood, T., Egger, J.: Augmented reality in support of industry 4.0—implementation challenges and success factors. Robotics and Computer-Integrated Manufacturing 58, 181–195 (2019), https://www.sciencedirect.com/science/article/pii/S0736584518304101
27. Masood, T., Egger, J.: Adopting augmented reality in the age of industrial digitalisation. Comput. Ind. 115(C) (feb 2020), https://doi.org/10.1016/j.compind.2019.07.002
28. Mengoni, M., Ceccacci, S., Generosi, A., Leopardi, A.: Spatial augmented reality: An application for human work in smart manufacturing environment. Procedia Manuf 17, 476–483, (2018), https://doi.org/j.promfg.2018.10.072
29. Merino, L., Schwarzl, M., Kraus, M., Sedlmair, M., Schmalstieg, D., Weiskopf, D.: Evaluating mixed and augmented reality: A systematic literature review (2009-2019 (2020-10), https://doi.org/10.1109/ISMAR50242.2020.00069.
30. Microsoft: HoloLens 2 — AR Headset. https://www.microsoft.com/en-us/hololens/hardware (2021), online; accessed 2021-07-06

31. Microsoft: Introducing instinctual interactions. https://docs.microsoft.com/en-us/windows/mixed-reality/design/interaction-fundamentals (2021), online; accessed 2021-07-06
32. Microsoft: Microsoft Mixed Reality Toolkit v2.4.0. https://github.com/microsoft/MixedRealityToolkit-Unity/releases/tag/v2.4.0 (2021), online; accessed 2021-07-06
33. Microsoft: Mixed Reality Dynamics 365 Guides (2021), https://dynamics.microsoft.com/en-us/mixed-reality/guides/, , accessed: 27 April 2021.
34. Microsoft: Overview of authoring a guide in Dynamics 365 Guides. https://docs.microsoft.com/en-us/dynamics365/mixed-reality/guides/authoring-overview (2021), online; accessed 2021-07-06
35. Nicolai, T., Sindt, T., Kenn, H., Witt, H.: Case study of wearable computing for aircraft maintenance. 3rd Int. Forum Appl. Wearable Comput (June), 1–12, (2006)
36. Nizam, S., Abidin, R., Hashim, N., Lam, M., Arshad, H., Majid, N.: A review of multimodal interaction technique in augmented reality environment. Int. J. Adv. Sci. Eng. Inf. Technol 8(4–2), 1460, (2018-09), https://doi.org/10.18517/ijaseit.8.4-2.6824
37. Palmarini, R., Erkoyuncu, J.A., Roy, R., Torabmostaedi, H.: A systematic review of augmented reality applications in maintenance. In: Robotics and Computer-Integrated Manufacturing, vol. 49 (2018), https://doi.org/10.1016/j.rcim.2017.06.002
38. P.T.C.: Vuforia expert capture (2021), https://www.ptc.com/en/products/augmented-reality/vuforia-expert-capture, accessed:
39. Quandt, M., Knoke, B., Gorldt, C., Freitag, M., Thoben, K.: General requirements for industrial augmented reality applications. Procedia CIRP 72, 1130–1135, (2018), https://doi.org/10.1016/j.procir.2018.03.061
40. Sanna, A., Manuri, F., Lamberti, F., Paravati, G., Pezzolla, P.: Using handheld devices to support augmented reality-based maintenance and assembly tasks. In: 2015 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas. pp. 178–179,. NV (2015), https://doi.org/10.1109/ICCE.2015.7066370
41. Sethi, A., Sethi, S.: Flexibility in manufacturing: A survey. Int. J. Flex. Manuf. Syst 2(4), 289–328, (1990), https://doi.org/10.1007/BF00186471
42. Souza Cardoso, L., Mariano, F., Zorzal, E.: A survey of industrial augmented reality. Comput. Ind. Eng 139, 106159, (2019-11), https://doi.org/10.1016/j.cie.2019.106159
43. Tainaka, K., Fujimoto, Y., Kanbara, M., Kato, H., Moteki, A., Kuraki, K., Osamura, K., Yoshitake, T., Fukuoka, T.: Guideline and tool for designing an assembly task support system using augmented reality. In: 2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR). pp. 486–497. IEEE Computer Society, Los Alamitos, CA, USA (nov 2020), https://doi.ieeecomputersociety.org/10.1109/ISMAR50242.2020.00077
44. Tang, A., Owen, C., Biocca, F., Mou, W.: Comparative effectiveness of augmented reality in object assembly. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. p. 73–80. CHI '03, Association for Computing Machinery, New York, NY, USA (2003), https://doi.org/10.1145/642611.642626
45. UnityTechnologies: Unity3d - 2019.4. https://unity3d.com/get-unity/download/archive (2021), online; accessed 2021-07-06
46. Uva, A., Gattullo, M., Manghisi, V., Spagnulo, D., Cascella, G., Fiorentino, M.: Evaluating the effectiveness of spatial augmented reality in smart manufacturing: a solution for manual working stations. Int. J. Adv. Manuf. Technol 94(1–4), 509–521, (2018), https://doi.org/10.1007/s00170-017-0846-4
47. Wang, X., Ong, S.K., Nee, A.: A comprehensive survey of augmented reality assembly research. Adv. Manuf 4(1), 1–22, (2016), https://doi.org/10.1007/s40436-015-0131-4

Traian Lavric is an industrial PhD student at Bosch France and Télécom SudParis. He received an Engineering Degree in Computer Science from Transylvania University Brașov and worked as R&D engineer at Télécom SudParis. His research interests include augmented and virtual reality, content authoring, human-computer interaction and interactive multimedia content.

Emmanuel Bricard is IT director of elm.leblanc, French leader in thermal comfort and part of Bosch group. Engineer and holder of an Executive MBA, he heads the R&D department "Innovation Center for Operations" that focuses on Industry 4.0 technologies, from IoT to augmented reality.

Marius Preda is associate professor at "Institut MINES-Télécom" and Chairman of the 3D Graphics group of ISO MPEG. He contributed to various ISO standards with technologies in the fields of 3D graphics, virtual worlds and augmented reality. He received a Degree in Engineering from POLITEHNICA Bucharest, a PhD in Mathematics and Informatics from University Paris V and an eMBA from IMT Business School, Paris.

Titus Zaharia received the Engineering degree in electronics and telecommunications and the M.Sc. degree from University POLITEHNICA, Bucharest, in 1995 and 1996, respectively, and the PhD in mathematics and computer science from University Paris V. He is a full professor in Télécom SudParis. His research interests include visual content representation methods, with 2D/3D compression, reconstruction, recognition, and indexing applications.

Received: October 15, 2021; Accepted: December 01, 2021.

CIP – Каталогизација у публикацији
Народна библиотека Србије, Београд

004

COMPUTER Science and Information Systems : the International journal /
Editor-in-Chief Mirjana Ivanović. – Vol. 19, No 2 (2022) - . – Novi Sad (Trg D. Obradovića 3): ComSIS Consortium, 2022 - (Belgrade : Sigra star). –30 cm

Polugodišnje. – Tekst na engleskom jeziku

ISSN 1820-0214 (Print) 2406-1018 (Online) = Computer Science and Information Systems
COBISS.SR-ID 112261644

Cover design: V. Štavljanin
Printed by: Sigra star, Belgrade