Computer Science and Information Systems 18(1):115–128 https://doi.org/10.2298/CSIS200515046P

A Dual Hybrid Recommender System based on SCoR and the Random Forest

Costas Panagiotakis[1], Harris Papadakis[2], and Paraskevi Fragopoulou[2]

[1] Department of Management Science and Technology
Hellenic Mediterranean University
72100 Agios Nikolaos, Crete, Greece
Tel.: +30-28410-91203
cpanag@hmu.gr
[2] Department of Electrical and Computer Engineering
Hellenic Mediterranean University
71004 Heraklion, Crete, Greece
Tel.: +30-2810-379119
adanar@hmu.gr, fragopou@ics.forth.gr *

Abstract. We propose a Dual Hybrid Recommender System based on SCoR, the Synthetic Coordinate Recommendation system, and the Random Forest method. By combining user ratings and user/item features, SCoR is initially employed to provide a recommendation which is fed into the Random Forest. The two systems are initially combined by splitting the training set into two "equivalent" parts, one of which is used to train SCoR while the other is used to train the Random Forest. This initial approach does not exhibit good performance due to reduced training. The resulted drawback is alleviated by the proposed dual training system which, using an innovative splitting method, exploits the entire training set for SCoR and the Random Forest, resulting to two recommender systems that are subsequently efficiently combined. Experimental results demonstrate the high performance of the proposed system on the Movielens datasets.

Keywords: recommender systems, synthetic coordinates, random forest.

1. Introduction

The explosive growth and variety of information available on the Web frequently overwhelm users and lead them to make poor choices. This problem is addressed by Recommender Systems (RS), that have become increasingly popular in guiding users to make more wise decisions [28]. Recommender Systems provide the degree of preference of a user for an item for a variety of entities such as e-shop items, web pages, news, articles, movies, music, hotels, television shows, books, restaurants, friends, etc.

A variety of techniques have emerged in the field of recommender systems. One of the main techniques is Similarity-based Collaborative Filtering (CF) [1,3], classified into user-based Collaborative Filtering and item-based Collaborative Filtering. CF is based on a similarity function that takes into account user preferences and outputs similarities for

* C. Panagiotakis and P. Fragopoulou are also with the Foundation for Research and Technology-Hellas (FORTH), Institute of Computer Science, 70013 Heraklion, Crete, Greece.

pairs of users. More specifically, the basic idea of user-based CF approaches is to detect a set of users who have similar favorite patterns to a given user (i.e., "neighbor" set of the user) and recommend to the user those items that others in its "neighbor" set like. While, item-based CF approaches recommend an item to a user based on other items with high correlations (i.e., "neighbor" set of the item).

In Dimensionality Reduction methods, each user or item is represented by a vector, where a user's vector is the set of his ratings for all items in the system. The sparsity of these vectors renders it difficult to identify correlations between user-item pairs. For this reason, Dimensionality Reduction techniques are employed, such as Singular Value Decomposition [6], Principal Component Analysis, Probabilistic Latent Semantic Analysis and Latent Dirichlet Allocation [17]. The Matrix Factorization method [13,11] that characterizes both items and users by vectors of latent factors inferred from item rating patterns, is also a Dimensionality Reduction technique. High correlation between item and user factors lead to recommendations.

In [23], the SCoR Recommender System was proposed. SCoR assigns synthetic coordinates (vectors) to users and items (nodes) as proposed in [6], but instead of using the dot product, SCoR uses the Euclidean distance between a user and an item in the Euclidean space, so that, when the system converges, the distance between a user-item pair provides an accurate prediction of that user's preference for the item. SCoR has several benefits, it is parameter free, thus does not require parameter tuning to achieve high performance, and is more resistant to the cold-start problem compared to other algorithms from the literature. The Vivaldi synthetic network coordinates algorithm, which lies at the back-end of SCoR, has been successfully applied to movie recommendation [23], personalized video summarization [19], detection of abnormal profiles in RS [18,20], community detection [22], and to the interactive image segmentation problem [21] providing high performance results compared to other state-of-the-art methods on public datasets.

Different architectures of artificial neural networks and deep learning methods have been found to be effective in several domains including computer vision, pattern recognition [27,7] and also in recommender systems [9]. These methods are powerful in processing unstructured multimedia data for feature representation learning like audio, text, image and videos. Convolutional Neural Networks (CNNs) [9] have been applied on the results of a preprocessing step e.g. the outer product between user and item ratings to obtain the 2D interaction map, in order to model the user item interaction patterns and to capture high-order correlations. Deep Hybrid Models for Recommendation integrates neural building blocks to formalize more powerful and expressive models. In [30] and [5], a CNN and an RNN based hybrid models for hashtag and citation recommendation are proposed, respectively.

In [24], the extended LSTM model with a higher-order interaction layer, proposed in [24], is able to handle data sparsity, includes a novel attention mechanism to reduce the burden of encoding the entire user history into a cell vector, and time aware input and forget gates to handle irregular time gaps between input interactions [24]. The three hidden layer-system proposed in [24], has been evaluated on the users from Twitter, Google Plus and YouTube. The authors used Tweets and Google Plus posts and YouTube videos either liked or added to playlists.

Random Forest algorithms [2] have been successfully employed in recommender systems [29,26]. In [29], the authors propose a framework that integrates three-way decision

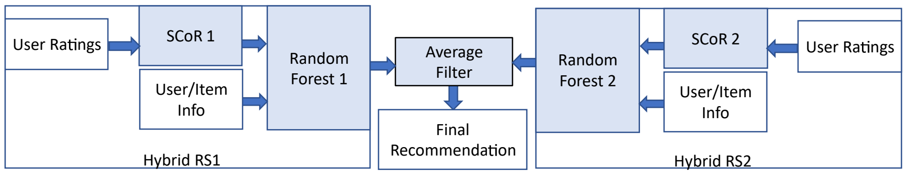


Fig. 1. The schema of the proposed dual hybrid system architecture.

and Random Forests to build recommender systems. Three way decision was introduced to map user recommendations for items, to "recommend", "not recommend", or "consult the user" actively for his/her preference. In [26], the authors propose a framework that employs reinforcement learning to derive good policies for Personalized Ad Recommendation (PAR) systems. Random Forest regression is used to efficiently learn a PAR policy.

There also exist hybrid methods that combine more than one approaches in order to improve recommendation accuracy. The system proposed in this paper belongs to this category. In [28], the UO-CRBMF model is combined with the IC-CRBMF model [14] to improve recommendation accuracy. In [25], a hybrid approach is proposed and applied to learning material. This hybrid system consists of attribute-based filtering and a genetic-based recommender system in order to improve the quality of recommendation in an e-learning environment. In [15], a Personalized Context-Aware Hybrid Travel Recommender System (PCAHTRS) is proposed, providing personalized tourist recommendations based on user ratings and their preferences. The hybrid recommendation algorithm employs user-based similarity, user's point-of-interest similarity, implicit user profiles and user's point-of-interest opinion similarity to predict users' ratings for tourist attractions.

In this paper, we propose a Dual Hybrid Recommender System by combining SCoR and the Random Forest approaches. SCoR receives user ratings and provides an initial recommendation to the Random Forest that gets as input user and item features to provide the final recommendation. A problem in such approaches is the use of smaller training sets in order to train both systems, which reduces the performance of each individual system. This problem is alleviated by the final proposed dual training system, resulting to two "equivileant" recommender systems that are efficiently combined. In addition, better results are obtained thanks to a novel method, proposed in this paper, that splits the training set into two "equivalent" parts for the training purposes of SCoR and the Random Forest, respectively. Figure 1 depicts the schema of the proposed dual hybrid system architecture. According to the proposed architecture, we have trained two SCoR systems (SCOR 1 and SCoR 2) and two Random Forest (Random Forest 1 and Random Forest 2) getting two recommendations. The final recommendation is given by the average of recommendations of Random Forest 1 and Random Forest 2.

The main contribution of this work is the improvement of the results of SCoR by efficiently combining context features and user ratings, while taking advantage of the Random Forest integration. The proposed approach, based on a dual process, also provides interesting directions towards alleviating two well-know problems in the field of recommender systems, namely, the cold-start and the data sparsity problems [12]. The user cold start problem appears in model-based methods, like SCoR [23] and Matrix Factorization

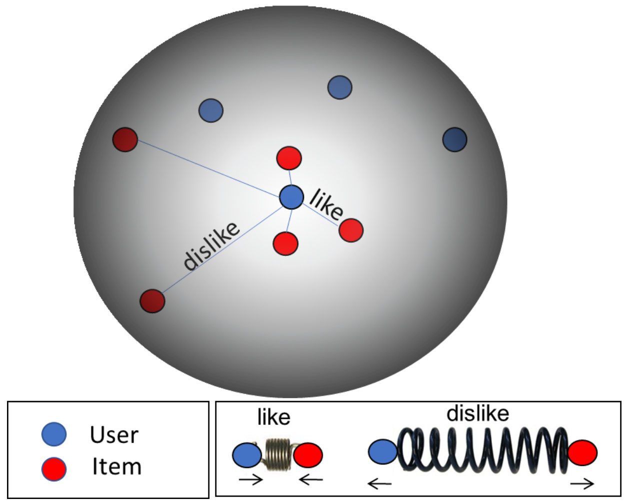


Fig. 2. A synthetic example following the execution of SCoR that shows the position of nodes (users and items) in $\mathbb{R}^2$. The item preferences for the user located in the center of the graph is indicated by the brightness of the graph background - from light grey (like) to darker grey (dislike).

[13], when a new user arrives and the system does not have user's historical behavior data. Data sparsity appears when several users have rated only a small subset of the items. The proposed system is highly flexible since any model-based method can be easily integrated by replacing SCoR. The selection of the remaining input features of Random Forest is also flexible, making the applicability of this work possible to any context where user ratings and user/item information are available.

The rest of the paper is organized as follows: Section 2 presents in detail the proposed dual hybrid recommend system. Section 3 describes the experimental setup along with the obtained results. Conclusions are provided in Section 4.

2. The Proposed Recommender System

2.1. SCoR

The proposed system (see Fig. 1) is based on SCoR, a novel personalized recommendation algorithm [23]. SCoR uses a Model-based Collaborating Filtering approach, which is dependent on a known set of user-to-item ratings, in order to train a preference prediction model. Thus, a number of preferences (ratings) of each user for some items must be already known. These are provided in the form of triplets (u, i, r), where r is the scalar rating of user u for item i.

In the core of SCoR lies the spring metaphor which inspired the Vivaldi synthetic network coordinate algorithm [4]. Essentially, the basis of SCoR is a Synthetic Euclidean Coordinate system, which randomly assigns a position in an N-dimensional Euclidean space to each element in the user U and the item I sets. The algorithm iteratively updates the positions of all elements (users and items) until, for every known rating (u, i, r), the Euclidean distance between user u and item i corresponds to the value r. The positions are updated using (1), as follows:

$$p(x) = p(y) + \delta \cdot (dd(x, y) - d(x, y)) \cdot b(x, y) \tag{1}$$

$$b(x, y) = \frac{p(x) - p(y)}{d(x, y)} \tag{2}$$

where $p(x)$, $p(y)$ are the positions of a user-item pair, $d(x, y)$ is their current Euclidean distance, $dd(x, y)$ is their desired distance (based on the rating value r). The unit vector $b(x, y)$ provides the direction towards which node x should move, and δ controls the method's convergence, since it is the fraction of distance node x is allowed to move toward its ideal position. Upon algorithm conversion, the Euclidean distance between user u and an unrated (by user u) item i provides a prediction for the preference of user u for item i. Thus, after the training phase, SCoR is able to provide a recommendation $\hat{r}(u, i)$ for any given user-item pair (u, i) in $O(1)$ based on the Euclidean distance between u and i. More details about SCoR can be found in [23].

Algorithm 1 shows the pseudo-code of the SCoR system. The input to the system is the set of users U, the set of items I and the values $minR$ (smallest rating), $maxR$ (highest rating). The training set TS and the test set VS consist of the given recommendations $(u, i, r(u, i)) \in TS$ and the predicted recommendations $\widehat{r}(u, i)$, with $(u, i) \in VS$ (produced by SCoR), respectively. $MSE(u)$ is the Mean Square Error of node u and its neighbors, while the procedure $getWeightedRandomSample$ selects nodes with smaller error more often for position updates. More details about SCoR can be found in [23].

Figure 2 shows a synthetic example after the execution of SCoR that shows the position of nodes (users and items) in $\mathbb{R}^2$. The distance between user u and item i corresponds to the predicted preference of u for item i. The item preferences for the user located in the center of the graph is indicated by the brightness of the graph background - from light grey (like) to darker grey (dislike).

2.2. Hybrid Recommender System

The proposed Hybrid Recommender System is based on the Random Forest approach, where the goal is to learn the recommendation for a given pair (u, i) taking as input: user's u and item's i information and the prediction $\hat{r}(u, i)$ of SCoR. In order to train the system, we have to use different training sets for SCoR and the Random Forest, otherwise low performance results are obtained (see *Hybrid RF* method in Section 3). The low performance is due to the fact that SCoR yields very low error for the instances of its training set, which is not the case for the test set. If we use the same training set for Random Forest, then it will give very high confidence to the input features provided by SCoR, making incorrect predictions on the test set.

In the proposed approach, the training set of ratings T is efficiently split into two "equivalent" parts (T_1 and T_2) to train SCoR and the Random Forest. Algorithm 2 presents the proposed splitting method described hereafter. Let G be the graph that shows the connections (ratings) between users and items in the training set (see Fig. 3(a)). The nodes of the graph are the union of users and items, and the edges of the graph correspond to the

Algorithm 1: The *SCoR* algorithm.

input : $i \in I, u \in U, (u, i, r(u, i)) \in TS, minR, maxR.$
output: $\widehat{r}(u, i)$

foreach $u \in U,$ **do**
    $p(u) =$ random position in $\mathbb{R}^n$
end
foreach $i \in I$ **do**
    $p(i) =$ random position in $\mathbb{R}^n$
end
repeat
    $(u, i) = getRandomSample(TS)$
    $[p(u), p(i)] = Vivaldi(p(u), p(i), r(u, i))$
until $\forall x \in I \cup U$ $p(x)$ *is stable*
foreach $(u, i) \in TS$ **do**
    $W(u, i) = e^{-0.2 \cdot MSE(u)} \cdot (dd(u, i) - d(u, i))^2$
end
repeat
    $(u, i) = getWeightedRandomSample(TS, W)$
    $[p(u), p(i)] = Vivaldi(p(u), p(i), r(u, i))$
until $\forall x \in I \cup U$ $p(x)$ *is stable*
foreach $(u, i) \in VS$ **do**
    $\widehat{r}(u, i) = maxR - (maxR - minR) \cdot \frac{||p(u)-p(i)||_2}{100}$
    $\widehat{r}(u, i) = min(max(\widehat{r}(u, i), minR), maxR)$
end

ratings in the training set. The splitting method tries to minimize the sum of the relative differences between the degree of a node (user or item) $x \in G$ in T_1 and T_2 (see Fig. 3(b)). The following function $f_{T_1,T_2}(x)$ shows the relative difference between the degrees of node x in T_1 $(deg_{T_1}(x))$ and T_2 $(deg_{T_2}(x))$:

$$f_{T_1,T_2}(x) = \frac{|deg_{T_1}(x) - deg_{T_2}(x)|}{\varepsilon + \min(deg_{T_1}(x), deg_{T_2}(x))}, \tag{3}$$

where ε is a small constant to prevent the zero in the denominator, e.g. $\varepsilon = 1$. When $f_{T_1,T_2}(x)$ is minimized, then $deg_{T_1}(x) = deg_{T_2}(x)$ or $|deg_{T_1}(x) - deg_{T_2}(x)| = 1$, which means that node x has almost the same number of edges in T_1 and T_2.

The proposed method is iterative and is based on sequential minimization. It starts from an initial "random" solution (see line 1 of Algorithm 2) and in every step it identifies the best pair of ratings (r_1 and r_2) that should be exchanged between T_1 and T_2 in order to minimize the following objective function $F(T_1, T_2)$: (see lines 3-16 of Algorithm 2).

$$F(T_1, T_2) = \sum_{x \in G} f_{T_1,T_2}(x) \tag{4}$$

The method terminates when there is no pair of ratings (r_1 and r_2) that can be exchanged between T_1 and T_2 which can further reduce the objective function (see line 17 of Algorithm 2). Following the execution of the proposed splitting method, a user u or an item v has almost the same number of connections (number of ratings) in T_1 and in T_2, while the total number of ratings in T_1 equals those in T_2.

Figure 3 depicts a synthetic example of a graph G and a splitting of its edges into sets T_1 and T_2. In this example, graph G consists of 8 nodes (3 users and 5 items) with 10

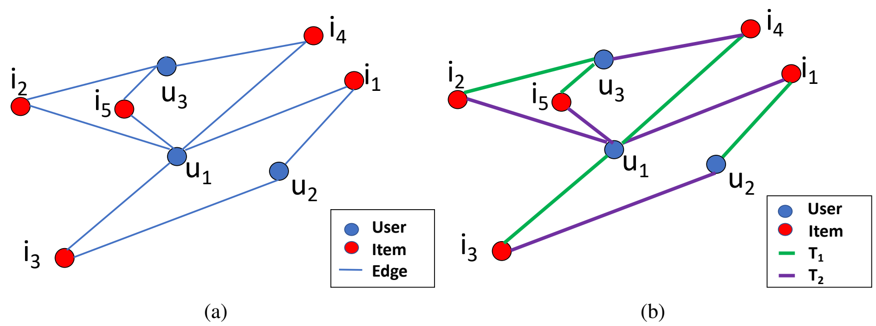


Fig. 3. A synthetic example of **(a)** a graph G and **(b)** the splitting of its edges into sets T_1 and T_2.

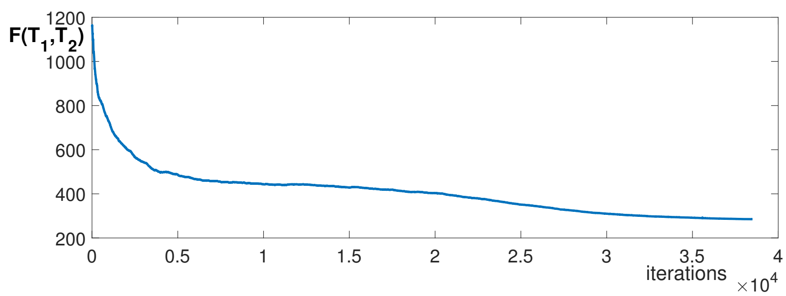


Fig. 4. The evolution of $F(T_1, T_2)$ on the *ML-100k* dataset.

edges (ratings). The splitting shown in 3(b) consists of two equivalent sets T_1 and T_2, each with 5 edges. In addition, each node has almost the same degree in both sets T_1 and T_2 according to the proposed splitting method. Figure 4 depicts the evolution of $F(T_1, T_2)$ on the *ML-100k* dataset. At convergence, it holds that $F(T_1, T_2)$ is not zero due to the existence of nodes with odd degree.

2.3. Dual Hybrid Recommender System

The two modules (SCoR and Random Forest) of the single Hybrid Recommender System presented in the previous subsection do not take advantage of the entire training set, since SCoR is only trained by T_1 and the Random Forest is only trained by T_2. The problem of reduced training for each individual module is alleviated by the proposed Dual Hybrid Recommender System.

The dual hybrid system is based on the dual training of two single Hybrid Recommender Systems (*Hybrid RS1* and *Hybrid RS2*). The training set of *Hybrid RS1* is used to train *Hybrid RS2* and vice versa, thus allowing each module to exploit the entire training set. Figure 5 depicts the proposed training schema. In *Hybrid* $RS1$, SCoR 1 is trained by T_1 and the Random Forest 1 by T_2. In *Hybrid* $RS2$, SCoR 2 is trained by T_2 and Ran-

Algorithm 2: The proposed splitting method.

input : T, G
output: T_1, T_2

$[T_1, T_2] = randomSplit(T)$
repeat
  foreach $r \in T_1$ **do**
    $[u, v] = getUserItem(G, r)$
    $c_1(r) = f_{T_1-\{r\},T_2\cup\{r\}}(u) - f_{T_1,T_2}(u) + f_{T_1-\{r\},T_2\cup\{r\}}(v) - f_{T_1,T_2}(v)$
  end
  foreach $r \in T_2$ **do**
    $[u, v] = getUserItem(G, r)$
    $c_2(r) = f_{T_1\cup\{r\},T_2-\{r\}}(u) - f_{T_1,T_2}(u) + f_{T_1\cup\{r\},T_2-\{r\}}(v) - f_{T_1,T_2}(v)$
  end
  $r_1 = argmin(c_1)$
  $r_2 = argmin(c_2)$
  if $c_1(r_1) + c_2(r_2) < 0$ **then**
    $T_1 = T_1 - \{r_1\}, \;\; T_2 = T_2 \cup \{r_1\}$
    $T_1 = T_1 \cup \{r_2\}, \;\; T_2 = T_2 - \{r_2\}$
  end
until $c_1(r_1) + c_2(r_2) < 0$

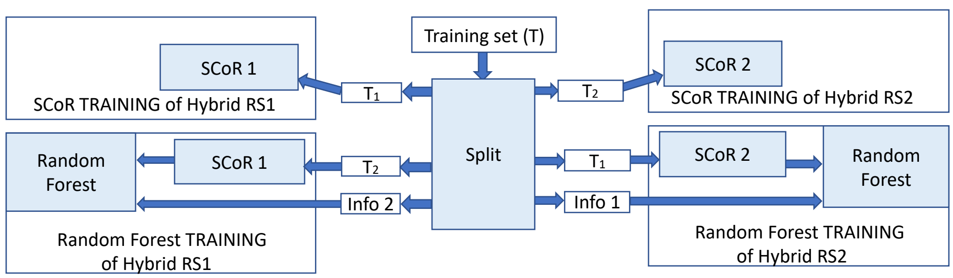


Fig. 5. The schema of the dual training system.

dom Forest 2 by T_1. Both systems have almost equal performance due to the "equivelant" training sets provided by the novel splitting method. Therefore, averaging the recommendations of the two equivalent systems to provide the final recommendation comes as a natural choice. This technique is illustrated in Figure 1.

3. Experimental Results

The experiments on the proposed *Dual Hybrid RS* method, are compared to the following three baseline methods

- *CF*: The user-based Collaborative Filtering approach with cosine similarity [1].
- *SCOR*: The method based exclusively on SCoR [23].
- *RF*: The method based exclusively on the Random Forest [2].

and the following two variants of the proposed method:

- *Hybrid RS*: The proposed single Hybrid Recommender System that uses SCoR and the Random Forest, both trained by the same entire training set.

– *Hybrid RS1 or Hybrid RS2*: The proposed single Hybrid Recommender System that uses SCoR and the Random Forest, each with half the training set as provided by the splitting method (T_1 for SCoR and T_2 for the Random Forest in $RS1$, and T_2 for SCoR and T_1 for the Random Forest in $RS2$).

3.1. Datasets and Features

The experiments are performed on the two well-known MovieLens datasets [8,23,29] with 5 rating-gradations (1-5) [3]:

– *ML-100k* consisting of 943 users and 1682 movies with 100,000 ratings.
– *ML-1M* consisting of 6040 users and 3883 movies with 1,000,000 ratings.

For the Random Forest, we use the following input features for each user and movie entry:

– *User entry*: average rating, number of ratings, age, gender and profession.
– *Movie entry*: average rating, number of ratings and genre.

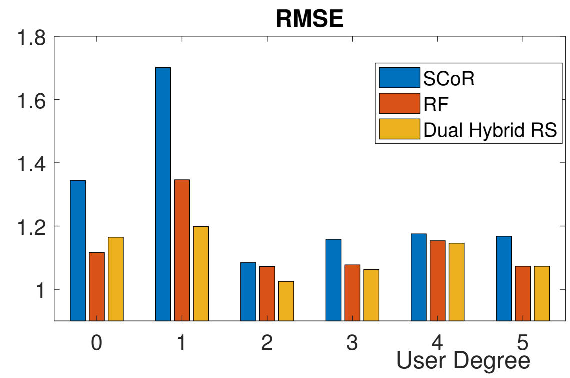
3.2. Performance Evaluation

Table 1. The $RMSE$ values for the proposed method and its variations on the *ML-100K* (left) and *ML-1M* (right) datasets.

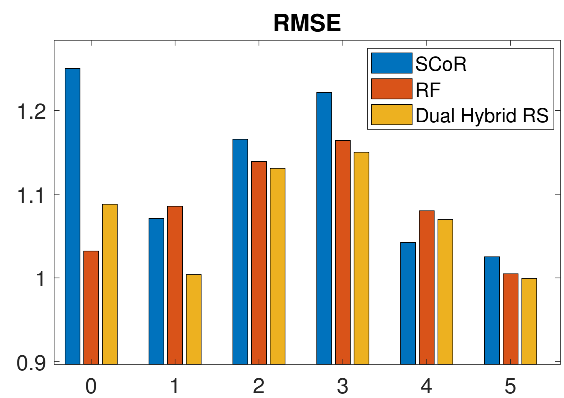
ML-100K		ML-1M	
METHOD	**RMSE**	**METHOD**	**RMSE**
CF	0.9522	CF	0.9508
SCOR	0.9474	SCOR	0.9576
RF	0.9450	RF	0.9551
Hybrid RS	0.9994	Hybrid RS	0.9853
Hybrid RS1	0.9517	Hybrid RS1	0.9765
Hybrid RS2	0.9569	Hybrid RS2	0.9782
Dual Hybrid RS	**0.9393**	**Dual Hybrid RS**	**0.9474**

The original dataset is randomly divided into training set (80%) and test set (20%). To evaluate the performance of the proposed method and its variations, we report the Root Mean Squared Error ($RMSE$) [10,1,23] for the test set. Table 1 presents the $RMSE$ values of the proposed method and its variations on the *ML-100K* and the *ML-1M* datasets. It is evident that the proposed method (*Dual Hybrid RS*) clearly outperforms all the remaining methods for both datasets. The second and third methods in performance are RF and $SCOR$, respectively, while the single hybrid (*Hybrid RS1* or *Hybrid RS2*) is the fourth method in performance due to reduced training. The worst results are obtained for method *Hybrid RS*, that uses the same training set for SCoR and the Random Forest.

[3] After the publication of this work, our intention is to make publicly available the code implementing the proposed method and the experiments performed.



(a)



(b)

Fig. 6. The $RMSE$ values for the proposed method and two baselines methods (SCoR and RF) on the **(a)** *ML-100K* and the **(b)** *ML-1M* datasets, as a function of the user degree.

3.3. Computational performance

The proposed system has been implemented using MATLAB, apart from SCoR which was implemented in Java. All experiments were performed on an Intel I7 CPU processor at 2.4 GHz. The processing time for training the *Dual Hybrid RS* for the *ML-100K* and *ML-1M* datasets are 70 secs and 25 mins, respectively. For the *ML-100K* dataset, it holds that 4%, 8% and 88% of the total processing time is consumed by the splitting method, SCoR and random forest, respectively. For *ML-1M*, it holds that 15%, 4% and 79% of the total processing time is consumed by the splitting method, SCoR and random forest, respectively. These results can be explained by the fact that the computational complexity of SCoR is $O(N)$, the complexity of the average case random forest training is $O(N \cdot$

$log^2 N)$ [16], while that of the splitting method is $O(N^2)$, where N denotes the number of training samples.

The processing time required for the execution of the *Dual Hybrid RS*, in order to predict the ratings of the test set for *ML-100K* and *ML-1M*, are 1.2 and 72 secs, respectively. This can be explained by the fact that the computational complexity of the pre-trained *Dual Hybrid RS* increases linearly with the size of the dataset, since it only uses two pre-trained random forests.

3.4. Cold Start and Sparsity Problems

This section examines the stability and efficiency of the proposed dual system with respect to the Cold Start and the Sparsity Problems [12], two well-known issues in recommender systems. The cold start problem occurs, when a new user or item enters the system. Sparsity appears for users that have rated only a small subset of the items, or items that have been rated by few users.

In order to examine the behavior of the proposed system with respect to the aforementioned problems, ratings from the training set are moved to the test set to ensure that there exists a minimum of users (e.g. 200 users) with zero or low degree (≤ 5) in the training set. Subsequently, we train the recommendation system and measure the $RMSE$ values in the test set, for the users as a function of the number of their ratings in the training set (user degree) as shown in Figure 6. It holds that for the cold start problem (zero degree users), as expected, SCoR fails to provide good recommendations, while the *RF* method and the proposed *Dual Hybrid RS* both yield satisfactory results. For the sparsity problem (users with small number of ratings), the proposed dual hybrid system yields slightly better results than RF and SCoR. This experiment demonstrates that the proposed method *Dual Hybrid RS* is a good combination of SCoR and the Random Forest that exploits the advantages of both systems and performs well on the Cold Start and the Sparsity Problems.

4. Conclusions

We presented a Dual Hybrid Recommender System based on the SCoR Recommender System and the Random Forest approaches. The proposed system efficiently combines context features and user ratings taking advantage of the Random Forest integration. In order to train the system, the training set is split into two "equivelant" parts, each one used to train one of the modules (SCoR or Random Forest) resulting to reduced training for both modules.

The proposed Dual Hybrid Recommender System improves the single training approach and it outperforms all the baseline methods and their variations as presented in our experimental results on the Movielens datasets. Furthermore, it performs well on the cold start and the sparsity problems. As future work, we plan to apply the proposed methodology to other datasets with richer context based features.

Acknowledgements. This research has been co-financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH - CREATE - INNOVATE (project code: T1EDK-02147).

References

1. Adomavicius, G., Kwon, Y.: Improving aggregate recommendation diversity using ranking-based techniques. IEEE Transactions on Knowledge and Data Engineering 24(5), 896–911 (2012)
2. Breiman, L.: Random forests. Machine learning 45(1), 5–32 (2001)
3. Cai, Y., Leung, H.f., Li, Q., Min, H., Tang, J., Li, J.: Typicality-based collaborative filtering recommendation. IEEE Transactions on Knowledge and Data Engineering 26(3), 766–779 (2013)
4. Dabek, F., Cox, R., Kaashoek, F., Morris, R.: Vivaldi: A decentralized network coordinate system. In: ACM SIGCOMM Computer Communication Review. vol. 34, pp. 15–26. ACM (2004)
5. Ebesu, T., Fang, Y.: Neural citation network for context-aware citation recommendation. In: Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval. pp. 1093–1096. ACM (2017)
6. Gorrell, G.: Generalized hebbian algorithm for incremental singular value decomposition in natural language processing. In: 11st Conference of the European Chapter of the Association for Computational Linguistics (2006)
7. Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., Lew, M.S.: Deep learning for visual understanding: A review. Neurocomputing 187, 27–48 (2016)
8. Harper, F.M., Konstan, J.A.: The movielens datasets: History and context. Acm transactions on interactive intelligent systems (tiis) 5(4), 19 (2016)
9. He, X., Du, X., Wang, X., Tian, F., Tang, J., Chua, T.S.: Outer product-based neural collaborative filtering. arXiv preprint arXiv:1808.03912 (2018)
10. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems 22(1), 5–53 (2004)
11. Hwang, W.S., Li, S., Kim, S.W., Lee, K.: Data imputation using a trust network for recommendation via matrix factorization. Computer Science and Information Systems 15(2), 347–368 (2018)
12. Jain, S., Grover, A., Thakur, P.S., Choudhary, S.K.: Trends, problems and solutions of recommender system. In: International Conference on Computing, Communication & Automation. pp. 955–958. IEEE (2015)
13. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer 42(8), 30–37 (2009)
14. Liu, X., Ouyang, Y., Rong, W., Xiong, Z.: Item category aware conditional restricted boltzmann machine based recommendation. In: International Conference on Neural Information Processing. pp. 609–616. Springer (2015)
15. Logesh, R., Subramaniyaswamy, V.: Exploring hybrid recommender systems for personalized travel applications. In: Cognitive informatics and soft computing, pp. 535–544. Springer (2019)
16. Louppe, G.: Understanding random forests: From theory to practice. arXiv preprint arXiv:1407.7502 (2014)
17. Mobasher, B., Burke, R.D., Sandvig, J.J.: Model-based collaborative filtering as a defense against profil injection attacks. In: The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA. pp. 1388–1393 (2006)
18. Panagiotakis, C., Papadakis, H., Fragopoulou, P.: Detection of hurriedly created abnormal profiles in recommender systems. In: International Conference on Intelligent Systems (2018)
19. Panagiotakis, C., Papadakis, H., Fragopoulou, P.: Personalized video summarization based exclusively on user preferences. In: European Conference on Information Retrieval (2020)
20. Panagiotakis, C., Papadakis, H., Fragopoulou, P.: Unsupervised and supervised methods for the detection of hurriedly created profiles in recommender systems. Machine Learning and Cybernetics (2020)

21. Panagiotakis, C., Papadakis, H., Grinias, E., Komodakis, N., Fragopoulou, P., Tziritas, G.: Interactive image segmentation based on synthetic graph coordinates. Pattern Recognition 46(11), 2940–2952 (2013)
22. Papadakis, H., Panagiotakis, C., Fragopoulou, P.: Distributed detection of communities in complex networks using synthetic coordinates. Journal of Statistical Mechanics: Theory and Experiment 2014(3), P03013 (2014)
23. Papadakis, H., Panagiotakis, C., Fragopoulou, P.: Scor: A synthetic coordinate based system for recommendations. Expert Systems with Applications 79, 8–19 (2017)
24. Perera, D., Zimmermann, R.: Lstm networks for online cross-network recommendations. In: IJCAI. pp. 3825–3833 (2018)
25. Salehi, M., Kamalabadi, I.N.: Hybrid recommendation approach for learning material based on sequential pattern of the accessed material and the learner's preference tree. Knowledge-Based Systems 48, 57–69 (2013)
26. Theocharous, G., Thomas, P.S., Ghavamzadeh, M.: Personalized ad recommendation systems for life-time value optimization with guarantees. In: Twenty-Fourth International Joint Conference on Artificial Intelligence (2015)
27. Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E.: Deep learning for computer vision: A brief review. Computational intelligence and neuroscience 2018 (2018)
28. Xie, W., Ouyang, Y., Ouyang, J., Rong, W., Xiong, Z.: User occupation aware conditional restricted boltzmann machine based recommendation. In: Internet of Things (iThings), 2016 IEEE International Conference on. pp. 454–461. IEEE (2016)
29. Zhang, H.R., Min, F.: Three-way recommender systems based on random forests. Knowledge-Based Systems 91, 275–286 (2016)
30. Zhang, Q., Wang, J., Huang, H., Huang, X., Gong, Y.: Hashtag recommendation for multimodal microblog using co-attention network. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia. pp. 3420–3426 (2017)

Costas Panagiotakis received the B.A, M.Sc., and Ph.D. degrees from the Dep. of Computer Science, University of Crete, Greece, in 2001, 2003, and 2007, respectively. Currently, he is Associate Professor and Head in Department of Management Science and Technology, Hellenic Mediterranean University and a researcher at the Institute of Computer Science (ICS), Foundation for Research and Technology-Hellas (FORTH) in Heraklion, Crete, Greece. He is the author of one book and more than 70 articles in international journals and conferences. His interests include image analysis, pattern recognition, multimedia and signal processing. For more information, please visit
https://sites.google.com/site/costaspanagiotakis/

Harris Papadakis received his B.Sc. and PhD in Computer Science from the University of Crete in 2001 and 2011 respectively, and the M.Sc. in Computer Science from the University of Patras in 2005. He is an Assistant Professor (under appointment) at the Department of Electrical and Informatics Engineering of the Hellenic Mediateranean University. He has worked in Large Scale Distributed Systems as well as Information Analysis. He has published over 40 papers related to the field of Data mining, Graph analysis, Community detection and Recommender systems as well as Distributed and P2P Systems.

Paraskevi Fragopoulou received her B.Sc. in Computer Science from the University of Crete in 1989, and the M.S. and Ph.D. degrees in Computer Science from Queen's University, Ontario, Canada in 1990 and 1995, respectively. Currently, she is a Professor of

Computer Science at the Department of Electrical and Computer Engineering, Hellenic Mediterranean University, Crete, Greece, and an Associated Researcher at the Institute of Computer Science, Foundation for Research and Technology-Hellas (FORTH-ICS) as member of the Distributed Computing Systems and Cybersecurity (DiSCS) Laboratory. She has co-authored more than 60 conference/journal papers and book chapters. Her primary research interests are in the areas of Distributed Algorithms, Internet Technology, Recommender Systems and Online Social Networks.

Received: May 15, 2020; Accepted: September 9, 2020.