Computer Science and Information Systems 21(3):923–946 https://doi.org/10.2298/CSIS230610014P

Visit Planner: A Personalized Mobile Trip Design Application based on a Hybrid Recommendation Model

Harris Papadakis[1], Costas Panagiotakis[2], Paraskevi Fragopoulou[1], Georgios Chalkiadakis[3], Errikos Streviniotis[3], Ioannis-Panagiotis Ziogas[3], Michail Koutsmanis[3], and Panagiotis Bariamis[4]

[1] Department of Electrical and Computer Engineering,
Hellenic Mediterranean University, Estavromenos, Heraklion, Crete, Greece
adanar@hmu.gr
fragopou@ics.forth.gr
[2] Department of Management Science and Technology,
Hellenic Mediterranean University, Lakonia, Agios Nikolaos, Crete, Greece
cpanag@hmu.gr
[3] School of Electrical and Computer Engineering,
Technical University of Crete, Kounoupidiana, Chania, Crete, Greece
gchalkiadakis@tuc.gr
estreviniotis@tuc.gr
iziogas@tuc.gr
mkoutsmanis@tuc.gr
[4] Netmechanics LLC,
Solonos 23, Heraklion, Crete, Greece
panos@netmechanics.gr

Abstract. The paper presents Visit Planner (ViP), a mobile application prototype that provides a solution to the challenging tourist trip design problem. ViP follows a holistic approach offering personalized recommendations for Points of Interest (POIs) based on preferences either explicitly collected by the application, or inferred by the users' ongoing interaction with the system. ViP proposes to the final user, a trajectory of POIs calculated using an Expectation Maximization method that maximizes user satisfaction taking into consideration a variety of time and spatial constraints for both users and POIs. Additionally, POIs are divided into categories, so that a certain number of POIs from each category to be included in the final itinerary. The application is implemented as a user-interactive system that allows the flexibility for easy content adaptation and facilitates management of content and services by the user. The prototype has been implemented for Android-based smartphones, on an open application environment, using standard communication protocols and open database technology. Currently, it is applied to the city of Agios Nikolaos in Crete, and is available for download from Google play.

Keywords: Mobile application, Recommendation system, Personalized tour itinerary, Expectation maximization Content-based, Collaborative-filtering

1. Introduction

In recent years, mobile applications designed for tourism have gained significant popularity. Since many tourists rely heavily on their mobile devices to enhance their travel experiences, the development of personalized mobile applications has become increasingly important. These applications offer targeted recommendations for POIs at a visiting site, which ultimately improve the cultural experience for the user [21, 6, 8, 9, 7]. This is particularly crucial for visitors who may have short stays in several locations, such as cruise ship tourists. Personalized mobile applications not only enhance the user's experience but also allow local communities and markets to promote their services and products in a more targeted manner, benefiting local economies. A recent comprehensive survey on the foundations and state of the art of mobile applications for tourism can be found in [23].

The designed mobile application prototype heavily relies on recommender algorithms to incorporate personalization into the final itinerary. Recommender systems aim to predict the preferences of users for items based on an analysis of preferences declared explicitly by the users or collected by the systems during previous use interactions [14]. Content-based recommender systems [15] rely solely on the user's declared preferences, while collaborative-based recommender systems [18] incorporate selections made by other users with similar preferences or similarity between items. At the same time, *Bayesian recommenders*, employ Bayesian updating of user models for efficient personalized recommendations [2, 1, 22].

The Visit Planner (ViP) application aims to provide users with the ultimate travel experience by offering personalized content and services. The application features a modern front-end that enables users to manage the content and services through an easy-to-use mobile application interface. The back-end of the designed prototype combines two innovative methods to provide users/visitors with the ultimate travel experience. The system uses a combination of four different approaches of personalized recommendation algorithms—collaborative filtering, content-based, and Bayesian—to extract from the POIs in the database those that best match the personalized interests of the traveler. After selecting the POIs that best match a user's interests using the recommender algorithms, an expectation maximization method is used to create a personalized itinerary recommendation that adheres to various real-time constraints such as opening and closing times of POIs, start/end time and other time constraints provided by the visitor, spatial locality of the POIs, budget constraints, preferred POI categories and locations, maximum trip duration, and other features such as preferred POI categories (e.g., travel, history).

This paper presents the functionality and the different components that comprise the current version of the Visit Planner App. The prototype has been currently applied to the Municipality of Agios Nikolaos in Crete, and is available though Google Play[5]. The ViP prototype can be adapted to any location by appropriately updating the content of its database module.

Among the main contributions of ViP is the fact that it incorporates an exhaustive variety of POI types, catering for every possible need and tase of the visotor (see Listing 1.3 in the Appendix). In addition, it is quite simple to use, with minimal participation from the user, while at the same time allowing for a large degree of control of the user over the created itinerary.

The remainder of the paper is organized as follows: Following the Introduction, Section 2 describes the related work. Section 3 presents the system requirements along with the user interface from the entering the user preferences and constraints to the derivation

[5] https://play.google.com/store/apps/details?id=com.netmechanics.vip

of the final personalized trip itinerary. The implementation and system architecture are presented in Section 4 including the front-end, back-end, database and middleware. Furthermore, in Section 4 the personalized recommendation algorithms used at the back-end of the application and the way they are invoked depending on each specific application scenario is described. Section 4 also presents the expectation maximization method that provides the final itinerary. We conclude in Section 5 with the main achievements and some directions for future research.

2. Related work

There is a significant number of publications on mobile applications for the tourism trip design problem, others focusing on making recommendations for POIs of interest, others employing data and user profile categorization and taxonomy data, while others related to the creation of personalized itineraries.

The authors of [16] elaborate on the challenges of designing apps for tourists through the evaluation of existing apps. They provides important implications for developers regarding application usability focusing on the parameters that guide the motivation of end users to use and reuse mobile apps that provide guided tours.

A systematic review of the personalized tourist trip design problem, also known as the orienteering problem in operations research is presented in [23]. The paper provides a review taxonomy and analyzes the main variants to the problem, the objectives, as well as the proposed solutions, an proposes threads for further research for solutions to new realistic problems. Another recent comprehensive survey on the foundations and state of the art of mobile applications for tourism can be found in [23].

In [21], a mobile application is developed that implements an algorithm for the Tourist Trip Design Orientering problem. The algorithm takes into consideration time dependencies, and analyzes in real time in combination the time constraints of the users and those of available POIs. The solution is based on a k-means algorithm, and is optimized using a genetic algorithm to improve the proposed tour itinerary. In order to facilitate recommendations, a parameterized fitness function is used to include any context element in the recommendation. The provided solution is scalable and adaptable to changes in the environment and in user preferences, thus offering a real time solution to the problem.

In [6], the authors present a hybrid planning service aiming to provide tourists with a sequence of attractions that interests them based on their previous interactions with the system. The service employs a model based recommendation system, named SCoR that operates on a synthetic coordinates principle. One of SCoR's main benefits is that it allows to incorporate additional training information on the fly without having to perform the training process from the beginning. The prototype has been implemented for Android-based smartphone and has been evaluated for St. Petersburg city. For the evaluation a database has been formed that includes attraction location information from the OpenStreeMaps platform, location description and media from Wikipedia, and ratings from Google Place.

In [8], the authors propose the ACUX recommendation system, in order to recommend POIs to visitors in a personalized manner. The proposed recommendation system replies on a collection of typologies in order to assign the visitors to one or more out of the eight available ACUX profiles. The classification is performed in order to capture the nonverbal preferences of visitors, and to provide them with personalized suggestions of potential POIs that match their preferences.

The authors of [9] present LOOKER, a mobile application for Android devices, that implements a content-based filtering recommendation system. The system relies on tourist related content collected from the users' social media posts, to make personalized suggestions. The back-end of the application implements a multilayer user profile approach, spiting services of different kinds into layers (restaurants, hotels, POIs) to infer the interests of travelers for available items.

Other papers deal with the creation of tourist guides using innovative mobile technology. One of the first efforts is presented in [7]. This paper presents the implementation prototype of a city tourist guide generation system for the city of Mytilene, the myMytileneCity guide. The service allows tourists to declare their preferences, and based on those, a custom application is created, that is downloadable to a mobile device. Following its installation the application is fully functional without connection. A push model allows users to be signaled when new content becomes available by the administrator.

The ViP application presented in this paper follows a holistic approach including personalized recommendations, but also offering an itinerary creation component that takes a multitude of spatial and time constraints into consideration. Recommendations are based on preferences declared by the users but also on users' ongoing interaction with the service. The application allows easy content adaptation and facilitates management of content and services by the user.

3. System Requirements

The main goal of the system is to provide users with meaningful suggestions and a pleasurable itinerary regarding their short visit to the city. To this aim, we have made the following system design choices.

1. Preference specification must be as short as possible while at the same time providing necessary information for accurate and meaningful suggestions.
2. Final selection of POIs must be done by the users, in a quick and simple fashion, in order to increase their satisfaction.
3. The final output of the system will be an itinerary which will include arrival and departure times for each included POI, while satisfying the constraints set by the users.

3.1. Usage

In order to use the system, the user must perform an one time registration process in order for their profile to be created. As seen in Fig. 1 this entails the provision of an email and password as well as a few demographic information, essential to parts of the recommendation algorithms. In the "Trip Type" drop-down menu, the user specifies whether they are travelling alone, with family or friends, children or a group.

After registration, the user will also be asked to specify his/her preferences, by being asked a small number of questions. This process is described in detail in the Section 3.1. After the completion of this process and/or after each subsequent successful login, the

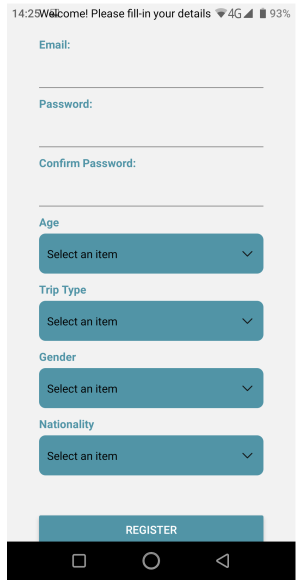

Fig. 1. New user registration form

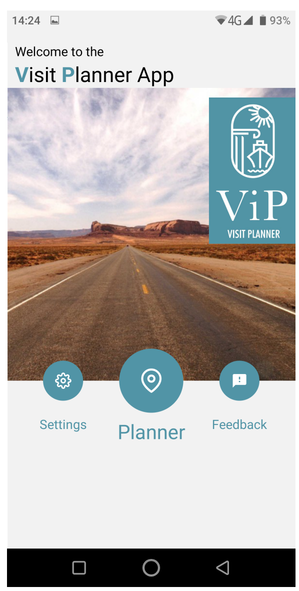

Fig. 2. Welcome screen

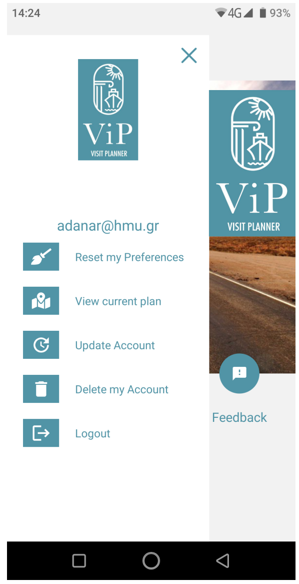

Fig. 3. Settings

user will be presented with the “Welcome” screen (Fig. 2). The available options on this screen include:

1. The “Settings” option which displays the “Settings” screen to the user, enabling them to Reset their preferences by repeating the “Specify preferences” procedure described in Section 3.1, View their most recent itinerary, Update their account by re-submitting the demographic information given during the Registration process, Delete their account from the system and finally Logout from the system.
2. The “Feedback” option which enables the user to write a textual review of the application
3. The “Planner” option where the user may begin the process of obtaining a new itinerary. This process is described in detail in Section 3.1.

Specifying preferences Depending in the recommendation algorithm that will be invoked for each user (see Section 4.4), we have implemented three different ways for the user to specify his/her preferences. Only one will be used for each user, depending on the assigned recommendation algorithm, during the registration process. However, the user can, at any time, request the repetition of the process in order to change his/her preferences (see the “Settings” menu in Section 3.1). In the first way of specifying preferences, the user is presented with a small number (10) of POI categories (not specific POIs, see Listing 1.3) and is asked to rate them (see Fig 4). In the second manner of preferences specification, the process is similar, with the only difference that the user is asked to simply specify whether they like or dislike each one of the POI categories they are presented (see Fig. 6). Finally, the third method, the user is presented with 5 different POI categories

Fig. 4. POI category rating screen

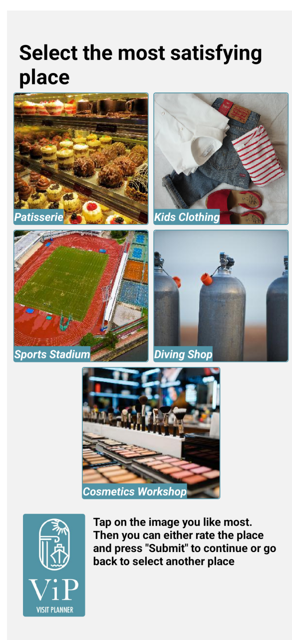

Fig. 5. POI category preference screen

Fig. 6. POI category like/dislike screen

and is asked to choose the most preferred one (see Fig. 5). They are then asked to rate that particular POI category as per the first method (see Fig. 4).

Obtaining itinerary The process of obtaining an itinerary is the main functionality of the system. The user initiates the process by selecting the "Planner" option from the "Welcome" screen. They are then presented with a form where they are able to specify the constraints of their desired itinerary (Fig. 7). The constraints include the visit starting time, the visit duration (available time), the expenditure level (expressed in 5 different levels) as well as whether their itinerary should include venues for food.

After specifying the desired constraints and selecting the "Create Itinerary" option, the recommender module of the system will be used in order to provide 20 POIs which best fit the specified preferences of the user (see Fig. 8). The user is then asked to select which ones will be included in the itinerary. This design choice was made in order to maximize the user satisfaction by asking him/her to participate in the final selection, in a simple and quick manner. During this selection process, the user is presented at the top of the screen with the remaining time left, given their current selections. In addition, during the selection process, some of the recommended POIs may be unavailable (greyed out) depending on their compatibility with the current selections. For instance, a POI may not be compatible due to its opening hours, or because its distance to the rest of the selected POIs is too big and thus it is not feasible to be visited in the specified time constraints. The user is, of course, able to also de-select POIs in order to for instance make time for other selections.

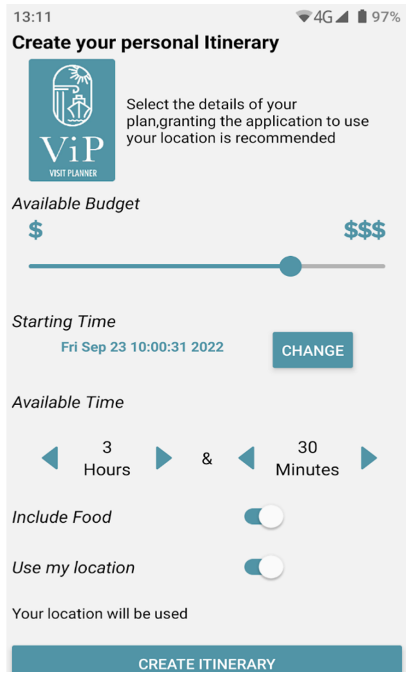

Fig. 7. Specify constraints screen

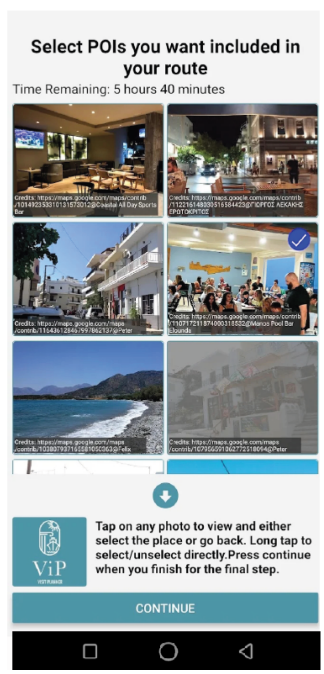

Fig. 8. POI selection

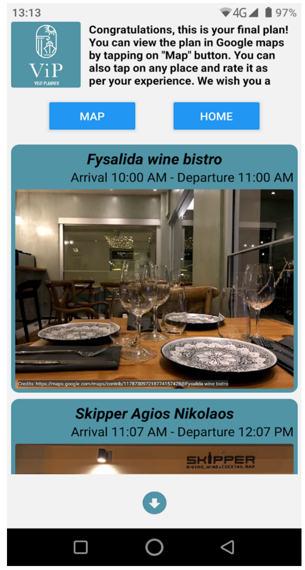

Fig. 9. Itinerary

After all selections have been made, the user will be presented with the final itinerary, using two different modes, as seen in Figs. 9 and 10. The first mode presents the sequence of the selected POIs in the most efficient order, as specified by the "Itinerary creation" module of the system, which is described in detail in Section 3.1. The mode also specifies to the user the arrival and departure times from each POI, in order for the user to be able to follow the created itinerary. Travel times by foot have been included in the computed times. Finally, the "Map" mode of the itinerary presents the user with the map of the area, where the location of the POIs that comprise the itinerary have been indicated in a numbered fashion, depending on their position in the itinerary sequence. The user is able to select each one of them and be transferred to the "Google Maps" application on their device in order to obtain the path to each POI.

It is important here to emphasize that the user is also able to rate any POIs that have been visiting by them, in order to further refine their preferences profile and obtain even more accurate POI recommendations.

4. System Architecture and Implementation

The architectural design and development of the platform should incorporate and implement the following key features:

1. Hardware should be available as a service (PaaS/SaaS).
2. Graded access depending on the type of services and the identity of the users.
3. Relational database

They should also ensure:

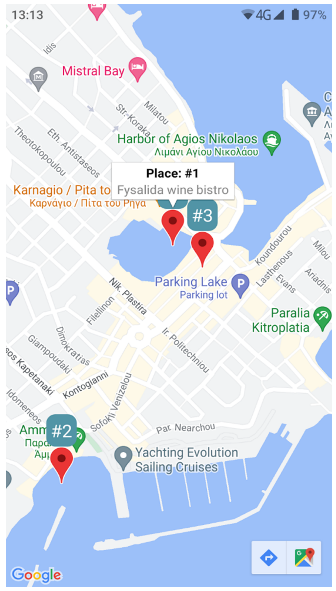

Fig. 10. Itinerary map

– An open application development environment.
– Standard communication protocols.
– Open environment in terms of database technology.
– Management of the content and services of the web portal and applications should be done by users through a simple web browser, allowing easy management & supervision.

The ViP pilot platform has been implemented with "Kubernetes" technology to be able to offer the software/platform as a service (SaaS/PaaS).

The architecture of the implemented prototype consists of four main components, illustrated in Fig. 11. These are:

1. The Front-end which is implemented as a mobile device application which interacts with the user.
2. The Back-end system which stores all necessary information.
3. The Middleware, whose purpose is to orchestrate the cooperation of the various components.
4. The Recommendations component which provides accurate and personalized POI recommendations, as well as specify the optimal selected POIs visit sequence through the Itinerary creation algorithm.
5. The Itinerary creation components that extracts the final tour trajectory.

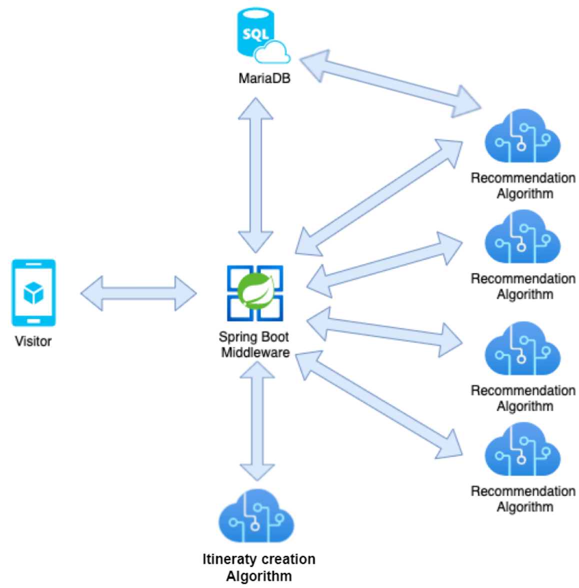

Fig. 11. System Architecture

4.1. Front-end

The front-end of the system is comprised of an android application which is available free of charge on Google Play[6].

4.2. Back-end

The Back-end of the system is comprised of a MariaDB relation database. The purpose of the database is to store all the necessary information for the smooth operation of the offered functionalities. It stores the following information, for several entities of the system:

1. User (see Listing 1.1 of the Appendix).
   The information stored for each user contains an id, their demographic information (nationality, gender, ageLevel, tripType), email as well as their specified preferences which were generated during the "Specifying preferences" procedure (see Section 3.1) as well as their ratings on specific POIs.
2. POI (see Listing 1.2 of the Appendix)
   The information stored for each POI contains its id, name, location, average rating, the categories it belongs to (see Listing 1.3), its opening hours, price Level, as well as the values of the POI's features and its users' ratings. The list of features is presented in the Listing 1.4 of the Appendix. For each POI, each one of the featured has been assigned a value between 0.0 and 1.0 which indicates the correspondence of the POI to that feature.

[6] https://play.google.com/store/apps/details?id=com.netmechanics.vip

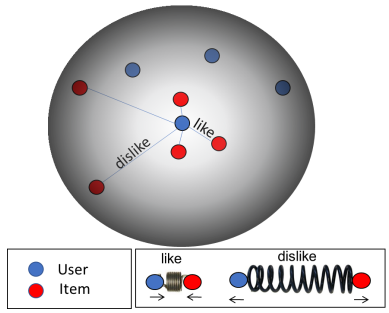

Fig. 12. A synthetic example after the execution of Synthetic Coordinates that shows the position of nodes (users and items) in $\mathbb{R}^2$. The distance between user u and item i corresponds to the prediction for the preference of user u for item i. The preference at each point of the graph for the user located in the center of the graph corresponds to the color brightness of the graph which varies from light grey (like) to dark gray (dislike)

4.3. Middleware

The Middle-ware is based on the well-known Spring Boot framework. Its main goal is to orchestrate the flow of control between the rest of the system's components, in order to implement the required functionality. It receives requests from the Front-end and serves them by contacting the rest of the modules, such as the various Recommendation algorithms that have been implemented as well as the database. All communication between components is performed through REST calls.

4.4. Recommendations component

The Recommendations module is comprised of four distinct, novel, recommender algorithms, which have been researched, developed and implemented for the needs of the system. These algorithms are summarily described in this Section. Each user is assigned one algorithm to be used in their case at all times. This is done in order to be able to evaluate the performance of each algorithm, based on the satisfaction of each user.

SCoR SCoR [13, 12] 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 (POIs) 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 (POI).

In the core of SCoR lies the spring metaphor which inspired the Vivaldi synthetic network coordinate algorithm [5]. 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 [13].

A synthetic example, after the computation of Synthetic Coordinates, is depicted in Figure 12 that shows the position of nodes (users and items). It depicts the preferences of the user located in the center of the graph and each item node of the graph via changes in the brightness of the background color varying from light gray (like) to dark grey (dislike).

Content-based Recommendations Using a Hierarchy Similarity Measure The second recommendation algorithm implemented in our system is a content-based one, that employs a hierarchy similarity measure on a well-defined hierarchy structure of POIs. Our hierarchy contains 90 generic POIs (i.e., each generic POI corresponds to one category) 430 real POIs as leafs, which belong to 90 categories, and correspond to touristic attractions of Agios Nikolaos. The categories were carefully selected given local and expert knowledge, and also the results of a survey we conducted and involved 150 real tourists visiting Agios Nikolaos. Moreover, we include in the hierarchy 90 "virtual" POIs corresponding to 90 generic images (one per category); these virtual POIs are employed for capturing user interests as follows.

In the beginning of the recommendation process, 15 out of 90 generic images (i.e., images that correspond to the generic POIs contained in our hierarchy tree) are presented to the user so as to be classified by her as "liked" or "disliked" (see Section 3.1 above). Our algorithm then computes the similarity between the "selected" generic images and all POIs, by using a modified version of a hierarchy similarity measure termed *extended Wu-Palmer similarity (XWP)* [17], which itself is an extended version of the so-called *Wu-Palmer similarity* [24]. In some detail, *XWP* takes into account the number of edges between the compared objects (i.e., POIs) and the hierarchy tree's root node, as well as the distance of the objects' Least Common Ancestor (or LCA) from the root node. Subsequently, we append the most similar POIs to the user's preferences from each generic image, and sample out a set of 20 POIs. This is the set provided to the user as the algorithm's final recommendations. The details of this approach can be found in [3].

Content-based Recommendations Using a Hybrid Similarity Measure This content-based approach takes as input data provided by the aforementioned algorithm in Section 4.4, and combines it with a novel *non-hierarchical* similarity metric, termed the

Weighted Extended Jaccard Similarity (WEJS) and introduced in [25]. As such, this approach constitutes a *hybrid* semantic similarity-based recommendation technique, which is employed to generate recommendations with respect to the user's interests.

In some detail, the WEJS metric takes into account only features of POIs which are equal in value. Each such feature is weighted based on the features of the user's preferences. WEJS is formally described in Eq. 3 below:

$$WEJS(X,Y) = \frac{\sum_{i \in X \cap Y} a * w_i}{\sum_{j \in X \cup Y} w_j}, a = \begin{cases} 1, & \text{if } v_i^X = v_i^Y \\ 0, & \text{otherwise} \end{cases}, \; with\; i \in X \cap Y \tag{3}$$

Here X and Y are the sets of the features of the two compared POIs; w_i and w_j are the weights of the ith member of the intersection of two compared sets and the jth member of their union, respectively; and v_i^X and v_i^Y denote the values of a given feature i for each set X and Y respectively, with $i \in X \cap Y$.

Assuming a maintained user profile built via the "likes" of generic POIs provided by the user (see Section 3.1 above), the use of *WEJS* on top of *XWP*-processed data, enables the focus to be placed on the actual user interests. Intuitively, *WEJS* considers a POI to be relevant to one matching the user profile if they *(a)* share the exact same characteristics (i.e., share features with the exact same values); and *(b)* these characteristics are deemed important (i.e., they are highly "weighted" by the user). The algorithm then recommends POIs that achieve high *WEJS* scores (with some exploration). The details of this approach can be found in [3].

Moreover, in [3] we extended this approach via putting forward a novel *hybrid recommender algorithm* that combines this semantic similarity-based recommender with a Bayesian Inference process to elicit user preferences. The latter will be further analyzed in the next Section.

A Bayesian Recommender Algorithm A fourth algorithm implemented in our system, is a Bayesian recommender method. To this end, every user and POI are modeled as multivariate normal distributions, based on the values of the features where POIs are concerned, whereas for users, the distributions are calculated based on questionnaires that were filled by actual tourist of Agios Nikolaos city. Notably, the preferences of each user are represented by a multivariate normal distribution and our recommender system does not posses such information. Thus, the goal of our system is to learn the preferences of each user, i.e., to generate a distribution that describes recommender's belief regarding each user's interests. For such purpose, we adopt the "*You are what you consume*" idea [1] in order to construct a representative model for each user's preferences.

To give some details, when a user enters our system, the recommender employs *Normal-Inverse Wishart (NIW)* conjugate priors to model its *beliefs* about the user's underlying parameters. NIW is a multivariate four-parameter family of continuous probability distributions, which is the *conjugate* prior of a multivariate Gaussian (normal) distribution with unknown mean and covariance matrix. Conjugacy ensures a closed-form computation of the posterior distribution. In our system, users—and also items—are modelled as *multivariate Gaussians*, hence the use of NIW priors ensures a computationally efficient Bayesian updating procedure for the user model, which is updated (in a Bayesian fashion) each time new preference information becomes available (for instance, whenever a user rates a POI that they visited) [1, 19].

Once our system has generated a distribution that describes the preferences of a specific user, it employs the Kullback-Leibler (KL) Divergence metric (see Eq. 4) in order to recommend the POIs that have the highest expected rating of a specific user.

Specifically, since both users and items share a common representation, as they are both modelled as *multivariate Gaussians*— we can employ the KL-divergence criterion in order to find "how similar" their distributions are. Formally, the KL-divergence between a Gaussian x and a Gaussian y, of dimension D each, is given by:

$$KL(y \parallel x) = \frac{1}{2}log|S_y^{-1}S_x| + \frac{1}{2}tr((S_y^{-1}S_x)^{-1}) - \frac{D}{2} + \frac{1}{2}(m_x - m_y)^T S_x^{-1}(m_x - m_y) \quad (4)$$

where S_y, m_y, S_x and m_x are the distributions' parameters, and $tr(\cdot)$ is the trace of the corresponding matrix [10]. In principle, a small KL-divergence between Gaussian x and Gaussian y implies similarity among the two, while a large KL-divergence implies they not similar. Thus, we assume that the more similar the distributions of a user u and an item i are, the higher user u rates item i. For more detailed description of the algorithm, please see [20, 19].

4.5. Itinerary creation

The proposed itinerary creation method is based on an user-interactive system that is depicted in Figs. 8, 9 and 10. The system also includes an expectation maximization technique that computes the best trajectory for the user, by maximizing an objective function that takes into consideration time, spatial, location, categories of POIs, and other constraints.

The itinerary creation is done in an iterative process so that in each step, the user selects the POI to be included in the visited set of the itinerary (see Fig. 8), from a set of unvisited and legal (valid) POIs that are automatically computed by the proposed method taking into account the user time budget and POIs opening hours and spatial constraints (user and POIs locations, etc). The ordering of the legal pre-mentioned POIs is provided to the user in a personalized way. Their sorting is done in descending order by its rating value according to the used recommender system described in Section 4.4. In each step of the method, the selected POI by the user is included to current itinerary in the position that maximizes a suitable objective function (see Eq. 5), reducing the system computational complexity. This objective function and the same methodology has been also successfully used in our recent work [11], where the itinerary was automatically created via a iterative generation process. The method terminates when the user time budget is exhausted, resulting the proposed itinerary as a timetable (Fig. 9). The user has also the option to view the itinerary in Google maps as depicted in Figure 10, respectively.

Next, we define preliminaries concerning the proposed methodology. We assume a graph (e.g. city map) with n POIs $P = \{p_1, ..., p_n\}$. Let T be the traveling time matrix ($n \times n$) of the pair-wise distances for all POI. Additionally, for each POI p_i the visit duration d_i and the opening time window o_i is known. According to the problem definition, the user provides the starting time st and the time budget B of the tour. This means that the tour itinerary should end at $st + B$ or earlier. s_i defines the gained user satisfaction per hour by visiting POI p_i. In our framework, s_i is computed offline by any of the three proposed recommendation methods described in Section 4.4. The output of our approach is an itinerary c, which defines the visited POIs as well as the corresponding temporal information. Therefore, an itinerary c is defined by a sequence of triples, where each triple (p_i, at_i, dt_i) is comprised by the visited POI p_i with the corresponding arrival at_i and departure dt_i times. Thus, we denote by $v(c)$, the sequence of triples (p_i, at_i, dt_i) of itinerary c, for which it holds that $dt_i > at_i$.

Hereafter, we describe the objective function that is used to evaluate an itinerary in order to find the best position of the new inserted POI. The proposed objective function F that has the following properties in order to to achieve the highest user satisfaction, while respecting the given problem constraints:

- For each POI (p_i) of itinerary c, F linearly increases with the corresponding gained user satisfaction per hour s_i that is multiplied by the visit duration $dt_i - at_i$. Intuitively, the larger gained satisfaction, the more preferable the itinerary c.
- The number of visited points $|v(c)|$ slightly increases the value of the objective function, so that when two legal itineraries yield almost the same user satisfaction, the larger itinerary will be more preferable.

The aforementioned properties are well captured by defining the objective function[7] $F(c)$ as following:

$$F(c) = (1 + log(|v(c)|)) \cdot \sum_{(p_i, at_i, dt_i) \in c} s_i \cdot (dt_i - at_i) \tag{5}$$

According to the proposed methodology, the expected value of the objective function $\overline{F}(c)$ is selected to be maximized as was done in [11] yielding more robust results.

$$\overline{F}(c) = \frac{B}{dt_n - at_1} \cdot F(c) \tag{6}$$

The expected value of the objective function of the itinerary c shows the upper limit of its current value $F(c)$ taking into account that the maximum time duration of an itinerary is at most B.

5. Conclusions

The Visit Planner (ViP), a mobile application prototype for personalized trip recommendations, has been presented. ViP recommends short tour itineraries of POIs based on the personalized preferences defined by tourists and their past interaction with the application. The proposed mobile application consists of several innovative modules, including the front-end web based interface and the back-end system architecture with its various components: the recommendation and the itinerary creation modules and the relational database. The trip design system is implemented based on an expectation maximization of a suitable objective function that takes into account time, spatial, location, categories of POIs, and other constraints. The itinerary creation process allows via a user-interactive system the flexibility to select from the recommended POIs those to be included in the final itinerary. The mobile application prototype has been implemented for Android-based

[7] The value of the objective function for non legal itineraries (according to the problem constraints) is $-\infty$ [11].

smartphones, on an open application environment, using standard communication protocols and open database technology.

The ViP prototype has been successfully implemented for the city of Agios Nikolaos in Crete, and is now available for download from Google Play. This mobile application prototype is the first of its kind for the tourist trip design problem applied to a city in Greece. Thanks to its modular and adaptable design, the proposed system can be effortlessly customized for any other city without requiring any changes to the system architecture. By simply incorporating new points of interest (POIs) and relevant city information such as opening hours, travel times, and city maps, the system can be tailored to fit any city. Apart from its direct application to other cities, we aim to focus on further developing and evaluating the Visit Planner App. Our plan is to extend the main functionality of the proposed system to include group itinerary recommendations [4]. The proposed system can be further enhanced by integrating information on public transportation such as buses, taxis, and metros into the itinerary creation process. Additionally, the system can incorporate details on accommodation and additional transportation options to create comprehensive trip recommendations for longer durations spanning several days across the entire region or sub-regions of Crete.

Acknowledgments. 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 B cycle (project code: T2EDK-03135).

Appendix

Listing 1.1. User schema

```
{
  "id": 0,
  "nationality": "string",
  "gender": "string",
  "ageLevel": "string",
  "tripType": "string",
  "email": "string",
  "unencryptedPassword": "string",
  "password": "string",
  "preferences": {
    "additionalProp1": 0,
    "additionalProp2": 0,
    "additionalProp3": 0
  },
  "reviews": [
    {
      "id": 0,
      "timestamp": "2023-02-14T11:25:38.933Z",
      "rating": 0,
      "userId": "string",
      "placeId": 0
    }
  ],
  "feedback": {
    "feedbackText": "string"
  }
}
```

Listing 1.2. POI schema


```
{
  "id": 0,
  "url": "string",
  "name": "string",
  "nameEn": "string",
  "latitude": 0,
  "longitude": 0,
  "rating": 0,
  "visitorsRating": 0,
  "categories": [
    "string"
  ],
  "openingHours": "string",
  "priceLevel": 0,
  "visitDuration": 0,
  "photoReference": "string",
  "photoAttribution": "string",
  "feature": {
    "culture": 0,
    "sunAndSea": 0,
    "historyArchaeology": 0,
    "adventureSports": 0,
    "affordablePrices": 0,
    "familyFriendlyActivitiesFacilities": 0,
    "ruralTourism": 0,
    "luxuryAccommodationLeisure": 0,
    "nightlife": 0,
    "gastronomyCuisine": 0,
    "generalShopping": 0,
    "localProductsShopping": 0
  },
  "visitorReviews": [
    {
      "id": 0,
      "timestamp": "2023-02-14T11:53:51.148Z",
      "rating": 0,
      "userId": "string",
      "placeId": 0
    }
  ]
}
```


Listing 1.3. POI categories

```
Bookstore
Library
Artwood Gallery
Artwork Gallery
Ceramics Gallery
Indoor Movie Theater
Outdoor Movie Theater
Archaeological Site
Historical Site
Monument
Archaeological Museum
Vintage Vehicle Museum
Historical Museum
Heritage Museum
Plant Museum
Chapel
Church
Historic Church
Monastery
Medieval Byzantine
Park
Square
Cape
Gorge
```

Grove
Hill
Mountain Village
Mountaintop
Bay
Non Organized Beach
Organized Beach
Bridge
Canal
Harbor
Island
Lake
Marina
River
Cafe Bar
Beach Bar
Cocktail Bar
Music Bar
Nightlife Club
Pool Bar
Pub
Rock Bar
Sports Bar
Wine Bar
Classic Cafeteria
Playground Cafeteria
Take Away Cafeteria
Tourist Cafeteria
Traditional Cafeteria
Bakery
Creperie
Ice Cream
Patisserie
Street Food
Asian Restaurant
Casual Greek Restaurant
Fine Greek Restaurant
Italian Restaurant
Greek Tavern
Mezedopoleio
Seafood Tavern
Flowers Shop
Cosmetic Store
Liquor Store
Diving Shop
Furs Store
Men Casual
Men Sports
Men Shoes
Women Casual
Women Sports
Women Shoes
Kids Clothing
Kids Shoes
Handmade Jewellery
Jewelry
Gift Shop
Traditional Products Shop
Ceramic Workshop
Cosmetics Workshop
Wood Workshop
Farm Tour
Boat Tour
Sports Stadium
Water Sports
Spa

Listing 1.4. POI Features

```
Culture
Sun&Sea
History/Archaeology
Adventure/Sports
Affordable Prices
Family Friendly Activities/Facilities
Rural Tourism
Luxury Accommondation/Leisure
Nightlife
Gastronomy/Cuisine
General Shopping
Shopping Local Products
```

References

1. Babas, K., Chalkiadakis, G., Tripolitakis, E.: You are what you consume: A bayesian method for personalized recommendations. In: Proceedings of the 7th ACM Conference on Recommender Systems. p. 221–228. RecSys '13, Association for Computing Machinery, New York, NY, USA (2013)
2. Barbieri, N., Costa, G., Manco, G., Ortale, R.: Modeling item selection and relevance for accurate recommendations: a bayesian approach. In: Mobasher, B., Burke, R.D., Jannach, D., Adomavicius, G. (eds.) 2011 ACM Conference on Recommender Systems, RecSys 2011, Chicago, IL, USA, October 23-27, 2011. pp. 21–28. ACM (2011)
3. Chalkiadakis, G., Ziogas, I., Koutsmanis, M., Streviniotis, E., Panagiotakis, C., Papadakis, H.: A novel hybrid recommender system for the tourism domain. Algorithms submitted for review (2023)
4. Chen, L., Cao, J., Chen, H., Liang, W., Tao, H., Zhu, G.: Attentive multi-task learning for group itinerary recommendation. Knowledge and Information Systems 63(7), 1687–1716 (2021)
5. Dabek, F., Cox, R., Kaashoek, F., Morris, R.: Vivaldi: A decentralized network coordinate system. ACM SIGCOMM Computer Communication Review 34(4), 15–26 (2004)
6. Kashevnik, A.M., Mikhailov, S., Papadakis, H., Fragopoulou, P.: Context-driven tour planning service: An approach based on synthetic coordinates recommendation. In: 24th Conference of Open Innovations Association, FRUCT 2019, Moscow, Russia, April 8-12, 2019. pp. 140–147. IEEE (2019)
7. Kenteris, M., Gavalas, D., Economou, D.: An innovative mobile electronic tourist guide application. Pers. Ubiquitous Comput. 13(2), 103–118 (2009)
8. Konstantakis, M., Christodoulou, Y., Aliprantis, J., Caridakis, G.: ACUX recommender: A mobile recommendation system for multi-profile cultural visitors based on visiting preferences classification. Big Data Cogn. Comput. 6(4), 144 (2022)
9. Missaoui, S., Kassem, F., Viviani, M., Agostini, A., Faiz, R., Pasi, G.: LOOKER: a mobile, personalized recommender system in the tourism domain based on social media user-generated content. Pers. Ubiquitous Comput. 23(2), 181–197 (2019)
10. Nielsen, F., Nock, R.: Emerging Trends in Visual Computing: LIX Fall Colloquium, ETVC 2008, Palaiseau, France, November 18-20, 2008. Revised Invited Papers. Springer Berlin Heidelberg, Berlin, Heidelberg (2009)
11. Panagiotakis, C., Daskalaki, E., Papadakis, H., Fragopoulou, P.: Personalized itinerary recommendation via expectation-maximization. In: 2022 IEEE 17th International Conference on Computer Sciences and Information Technologies (CSIT). pp. 210–213. IEEE (2022)
12. Panagiotakis, C., Papadakis, H., Papagrigoriou, A., Fragopoulou, P.: Improving recommender systems via a dual training error based correction approach. Expert Systems with Applications 183, 115386 (2021)
13. Papadakis, H., Panagiotakis, C., Fragopoulou, P.: Scor: A synthetic coordinate based system for recommendations. Expert Systems with Applications 79, 8–19 (2017)

14. Papadakis, H., Papagrigoriou, A., Panagiotakis, C., Kosmas, E., Fragopoulou, P.: Collaborative filtering recommender systems taxonomy. Knowl. Inf. Syst. 64(1), 35–74 (2022)
15. Pasquale Lops, M.d.G., Semeraro, G.: Content-based recommender systems: State of the art and trends. In: Recommender Systems Handbook, pp. 73–106. Springer-Verlag, Berlin, Heidelberg (2010)
16. Podsukhina, E., Smith, M.K., Pinke-Sziva, I.: A critical evaluation of mobile guided tour apps: Motivators and inhibitors for tour guides and customers. Tourism and Hospitality Research 22(4), 414–424 (2022)
17. Shenoy, M.K., Shet, K., Acharya, U.D.: A new similarity measure for taxonomy based on edge counting. International Journal of Web & Semantic Technology 3(4), 23 (2012)
18. Shi, Y., Larson, M., Hanjalic, A.: Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. ACM Comput. Surv. 47(1), 3:1–3:45 (May 2014), http://doi.acm.org/10.1145/2556270
19. Streviniotis, E., Chalkiadakis, G.: Multiwinner election mechanisms for diverse personalized bayesian recommendations for the tourism domain. In: Neidhardt, J., Wörndl, W., Kuflik, T., Goldenberg, D., Zanker, M. (eds.) Proceedings of the Workshop on Recommenders in Tourism (RecTour 2022) co-located with the 16th ACM Conference on Recommender Systems (RecSys 2022), Seattle, WA, USA and Online, September 22, 2022. CEUR Workshop Proceedings, vol. 3219, pp. 65–82. CEUR-WS.org (2022)
20. Streviniotis, E., Chalkiadakis, G.: Preference aggregation mechanisms for a tourism-oriented bayesian recommender. In: Aydogan, R., Criado, N., Lang, J., Sánchez-Anguix, V., Serramia, M. (eds.) PRIMA 2022: Principles and Practice of Multi-Agent Systems - 24th International Conference, Valencia, Spain, November 16-18, 2022, Proceedings. Lecture Notes in Computer Science, vol. 13753, pp. 331–346. Springer (2022)
21. Tenemaza, M., Luján-Mora, S., de Antonio, A., Ramírez, J.: Improving itinerary recommendations for tourists through metaheuristic algorithms: An optimization proposal. IEEE Access 8, 79003–79023 (2020)
22. Tripolitakis, E., Chalkiadakis, G.: Probabilistic topic modeling, reinforcement learning, and crowdsourcing for personalized recommendations. In: Pacheco, N.C., Carrascosa, C., Osman, N., Inglada, V.J. (eds.) Multi-Agent Systems and Agreement Technologies - 14th European Conference, EUMAS 2016, and 4th International Conference, AT 2016, Valencia, Spain, December 15-16, 2016, Revised Selected Papers. Lecture Notes in Computer Science, vol. 10207, pp. 157–171. Springer (2016)
23. W. Wörndl, D.H.: Mobile applications for e-tourism. Handbook of e-Tourism, Springer (2020, Springer)
24. Wu, Z., Palmer, M.: Verb semantics and lexical selection. arXiv preprint cmp-lg/9406033 (1994)
25. Ziogas, I.P., Streviniotis, E., Papadakis, H., Chalkiadakis, G.: Content-based recommendations using similarity distance measures with application in the tourism domain. In: 12th SETN Conference on Artificial Intelligence. pp. 1–10 (2022)

Harris Papadakis is an Assistant Professor of the Department of Electrical and Informatics Engineering of the Hellenic Mediterranean University in the field of "Distributed Services and Networks". He holds a PhD from the Department of Computer Science of the University of Crete, a degree in Computer Science from the Department of Computer Science of the University of Crete and a postgraduate MSc diploma from the Department of Computer Engineering of the University of Patras. He has participated as researcher and head in several projects.

Costas Panagiotakis is Associate Professor, Director of Data Science, Multimedia and Modelling Laboratory (DataLab) and Head of the Department of Management Science and Technology, HMU. He has been involved in several R&D projects funded by the EC during the last 23 years in the fields of computer vision, pattern recognition and recommender systems. He is also a researcher at the Computational Vision and Robotics Laboratory, Institute of Computer Science, Foundation for Research and Technology-Hellas.

Paraskevi Fragopoulou obtained her BA from the Computer Science Department of the University of Crete in 1989 and M.Sc. and Ph.D. in computer science in 1991 and 1995, respectively, from the Department of Computing and Information Sciences, Queen's University, Kingston, Ontario, Canada. She is a Professor at the Department of Electrical and Informatics Engineering of the Hellenic Mediterranean University (EL.ME.PA). She has worked in research positions at various European Research Institutes. She is a collaborating faculty member at the Institute of Technology and Research (ITE), Heraklion, Crete.

Georgios Chalkiadakis joined the Technical University of Crete in March 2011. Before coming to TUC, he was a Research Fellow at the School of Electronics and Computer Science, University of Southampton, UK, from December 2006 to January 2011. Georgios received his PhD from the University of Toronto, Canada, in 2007. His PhD thesis was a nominee for the IFAAMAS Victor Lesser Distinguished Dissertation Award (2007). Georgios has also worked as a software engineer at the Institute of Computer Science of the Foundation for Research and Technology - Hellas.

Errikos Streviniotis completed his undergraduate studies and acquired a diploma degree at the School of Electrical and Computer Engineering at the Technical University of Crete (December 2020). He acquired an M. Sc. in Electronic and Computer Engineering at the Technical University of Crete (December 2022) under the supervision of Professor Georgios Chalkiadakis. Currently, Errikos is a PhD candidate at the same institution under the supervision of Assistant Professor Nikos Giatrakos. In parallel to his studies, Errikos works as a research associate at the 'Athena' Research Center. His primary research focuses on Artificial Intelligence, Machine Learning, and Game Theory.

Ziogas Ioannis Panagiotis completed his undergraduate studies and acquired a diploma degree at the School of Electrical and Computer Engineering at the Technical University of Crete (February 2020). Acquired the M. Sc. in Research in Electronic and Computer Engineering at the Technical University of Crete (March 2023) under the supervision of Professor Georgios Chalkiadakis.

Michail Koutsmanis is currently a Ph.D. student at the School of Electrical and Computer Engineering in Technical University of Crete (since August 2022) under the supervision of Associate Professor Georgios Chalkiadakis. He earlier completed his undergraduate studies and acquired a diploma degree at the Department of Electrical and Computer Engineering in Aristotle University of Thessaloniki (July 2021). His research interests lie mainly on Bounded Rationality and Game Theory along with their implementation in Multi-agent Systems.

Panagiotis Bariamis received his BSc from the Technological and Educational Institute of Athens, in Information Technology. He is a full-time software engineer since 2003, currently working for Netmechanics S.A.

Received: June 10, 2023; Accepted: February 15, 2024.