



Article

A Novel Hybrid Recommender System for the Tourism Domain

Georgios Chalkiadakis ^{1,*}, Ioannis Ziogas ¹, Michail Koutsmanis ¹, Errikos Streviniotis ¹, Costas Panagiotakis ^{2,*} and Harris Papadakis ³

¹ School of Electrical and Computer Engineering, Technical University of Crete, 73100 Chania, Greece

² Department of Management Science and Technology, Hellenic Mediterranean University, 72100 Agios Nikolaos, Greece

³ Department of Electrical and Computer Engineering, Hellenic Mediterranean University, 71500 Heraklion, Greece

* Correspondence: gchalkiadakis@tuc.gr (G.C.); cpanag@hmu.gr (C.P.)

Abstract: In this paper, we develop a novel hybrid recommender system for the tourism domain, which combines (a) a Bayesian preferences elicitation component which operates by asking the user to rate generic images (corresponding to generic types of POIs) in order to build a user model and (b) a novel content-based (CB) recommendations component. The second component can in fact itself be considered a hybrid among two different CB algorithms, each exploiting one of two semantic similarity measures: a hierarchy-based and a non-hierarchy based one. The latter is the recently introduced *Weighted Extended Jaccard Similarity* (WEJS). We note that WEJS is employed for the first time within a recommender algorithm. We incorporate our algorithm within a real, already available at Google Play, tour-planning mobile application for short-term visitors of the popular touristic destination of Agios Nikolaos, Crete, Greece, and evaluate our approach via extensive simulations conducted on a real-world dataset constructed for the needs of the aforementioned mobile application. Our experiments verify that our algorithms result in effective personalized recommendations of touristic points of interest, while our final *hybrid* algorithm outperforms our exclusively content-based recommender algorithms in terms of recommendations accuracy. Specifically, when comparing the performance of several *hybrid* recommender system variants, we are able to come up with a “winner”: the most preferable variant of our *hybrid* recommender algorithm is one using a *(four elicitation slates, six shown images per slate)* pair as input to its Bayesian elicitation component. This variant combines increased precision performance with a lightweight preferences elicitation process.

Keywords: hybrid recommender systems; Bayesian recommenders; content-based recommendations; personalized recommendations; tourism recommenders; semantic similarity measures



Citation: Chalkiadakis, G.; Ziogas, I.; Koutsmanis, M.; Streviniotis, E.; Panagiotakis, C.; Papadakis, H. A Novel Hybrid Recommender System for the Tourism Domain. *Algorithms* **2023**, *16*, 215. <https://doi.org/10.3390/a16040215>

Academic Editors: Vicente García-Díaz and Edward Rolando Núñez-Valdez

Received: 17 February 2023

Revised: 13 April 2023

Accepted: 20 April 2023

Published: 21 April 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Tourism is a widespread activity and a huge industry in our time and era. According to the United Nations' World Tourism Organization, more than 900 million tourists travelled internationally in 2022 alone. In general, tourism can have a great impact on various aspects of social and economic life—including local infrastructure, transportation, land use, housing prices and even local lifestyle and associated entrepreneurship activities [1–3]. In their travels, tourists have to select among numerous alternatives of landmarks, places and in general *points of interest* (POIs) they can visit. This can be challenging for travellers that have no prior experience of or “inside knowledge” regarding their destination, and even more so for short-term visits. The exhaustive examination of all the possibilities is clearly impossible, thus an information system that can assist them in narrowing their options down to a more tight set would be of much use to tourists. As a result, a booming industry of travel-related *recommender systems* (RSs) [4] has been developed, in order to provide users with recommendations most relevant to their interests. Such systems are embedded in almost every modern mobile tour guide and similar applications.

Recommender systems are divided into three main categories with respect to the methods they adopt, namely the *content-based* (CB), *collaborative filtering* (CF) and *hybrid RSs* [4–6]. Nevertheless, they can further be grouped into additional categories, such as *Bayesian recommenders*, which employ Bayesian updating of user models for efficient personalized recommendations [7–9]. A recent research work on *travel recommenders* [10] has classified these into three major categories: hotel, restaurant and tourism recommenders. The latter can be related to tour planning, group recommendations, touristic attractions-related packages or travel packages. Recently, Bayesian recommender algorithms have been combined with social choice-theoretic methods to enable personalized and fair recommendations of touristic POIs, both in single-user and group recommendation environments [11,12].

Now, although *semantic similarity measures* have been mainly exploited in the domains of text analysis [13] and natural language processing [14], they have also been employed to correlate the preferences of tourists with touristic attractions. All semantic similarity measures describe the relations between different ontologies, and are classified into two categories [15]. The first corresponds to those that generate a hierarchical structure in order to measure similarity and are termed *hierarchy similarity measures*, while the second correlates ontologies without constructing a hierarchy tree and are termed *non-hierarchy similarity measures*.

Against this background, in this paper we put forward a novel recommender algorithm that is a *hybrid* of two approaches: a novel content-based (CB) recommender utilizing semantic similarity measures, and a Bayesian one. In fact, our CB recommender algorithm is itself, in some sense, a “*hybrid*” of two (CB) approaches that employ two types of semantic similarity measures—a hierarchy and a non-hierarchy one.

The hierarchy similarity measure of choice is the *Extended Wu–Palmer (XWP)* similarity, a variation of the metric proposed in [16], in which POIs lie on the nodes of a hierarchy tree, whereas the other metric is a novel non-hierarchy similarity measure called *Weighted Extended Jaccard Similarity (WEJS)* measure. WEJS calculates the correlation between different POIs by taking into account both the user’s preferences and the POIs’ features values. Thus it produces personalized recommendations. Subsequently, our CB algorithm combines these two measures into a *hybrid* (the term hybrid here denotes a combination of two similarity measures and it should not be confused with the hybrid recommender algorithm that is the final outcome of our work) similarity measures-based method. Finally, this CB algorithm is combined with a Bayesian Inference component that is responsible for building a user model via a lightweight Bayesian elicitation process that asks the user to rate *generic* images (corresponding to generic types of POIs). Then, a set of generic POI items corresponding to the user preferences captured by the Bayesian component are fed into the CB algorithm that eventually recommends the actual POIs that are most similar to the constructed user model. The high-level structure of our novel *Hybrid RS* is presented in Figure 1.

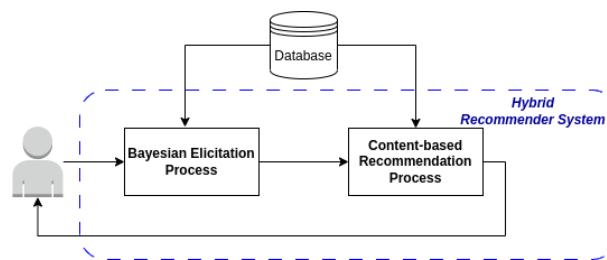


Figure 1. The high-level structure of our Hybrid Recommender System.

As such, the eventual *Hybrid RS* we propose has two desired properties that a simple Bayesian or a simple CB RS could not exploit. Specifically, the usage of a probability-density-based (usually a multi-dimensional Gaussian distribution) representation [17] along with *Bayesian Inference* provides us with a formal way to model the high uncertainty that exists in such settings [18], while the CB technique enables our system to utilize data

regarding (i) hierarchical structures that contain the available POIs; (ii) the “distance” between POIs’ characteristics and users’ personal interests, via exploiting information contained in user-specific weights assigned to different features.

Now, a common issue arising in recommender systems and especially in Collaborative Filtering techniques is the well-known *cold-start problem*. This refers to the inability of algorithms to provide meaningful recommendations due to the lack of ratings when a new user or item to recommend enters the system [19,20]. This can have a great impact on users’ experience when using travel planning applications related especially to short-term travels, since almost all users are new and presumably interact with the application only a handful of times. Therefore, the (short-term) efficiency of recommender algorithms incorporated in such applications is of utmost importance. In our case, we effectively circumvent and tone down the impact of the cold-start problem, by not employing a CF approach (i.e., by not relying on and comparing with others’ ratings) for producing personalized recommendations—but by using a Bayesian updating approach to elicit and maintain user profiles. Specifically, in our work we directly elicit new users’ preferences by presenting generic images to them, maintain their profiles as explained in the paper, and make recommendations based on their elicited interests. Avoiding the cold-start problem can be viewed as an advantage of our recommendation approach especially in the context of short-term tourist visits. Of course, quickly arriving at accurate user profiles is a challenge, which can be dealt with effectively via the use of appropriate easy-to-maintain priors [8,9,11,12].

We evaluate our *Hybrid RS* and its components using a real-world dataset from a popular tourist destination, the city of Agios Nikolaos in Crete, Greece. The data related to user preferences were collected by means of a survey answered by actual tourists, while data on POIs were collected via close collaboration with the local municipality, other sources of local knowledge and web sources. Our datasets and algorithms were constructed to be utilized within a real-world mobile app that offers personalized recommendations and tour scheduling services to short-term visitors of Agios Nikolaos and its vicinity. The app (see Figure 2), developed in close partnership with the city’s municipal authority, incorporates several recommender algorithms and a deterministic method for the recommendation of personalized itineraries [21], and is currently available at Google Play Store (<https://play.google.com/store/apps/details?id=com.netmechanics.vip&pri=1>, accessed on 17 February 2023).

The experiments we conducted verify the excellent performance of our RS algorithms, which regularly exhibit a precision of over 83%. In some detail, while our CB recommender components are shown to be quite effective in producing accurate personalized recommendations, our final Hybrid RS demonstrates a superior ability to extract the users’ profiles and utilize them in its operation. In particular, its use is shown to result in even more precise personalized recommendations than those generated by the CB recommenders, and it does so exploiting a *lightweight* Bayesian elicitation process that requires only a low number of interactions with a user in order to “learn” her profile.

In a nutshell, our work in this paper results in several contributions:

1. We employ for the first time within a recommender algorithm a novel non-hierarchy similarity measure, the *Weighted Extended Jaccard Similarity (WEJS)* (originally introduced in [22]).
2. We combine it with a hierarchy-based similarity measure to give rise to a novel hybrid CB recommendations algorithm.
3. We combine that CB algorithm with a Bayesian component to put forward a novel *Hybrid Recommender System*.
4. We evaluate our recommender algorithms via extensive simulations conducted on a real-world dataset, with results testifying to their effectiveness in producing accurate personalized recommendations.
5. Last but not least, our algorithm is already incorporated in a real-world tour planning mobile app for short-term visitors of a popular touristic destination.

The rest of this paper is structured as follows. In Section 2 we present background and briefly review related work. Section 3 is devoted to a detailed description of our CB recommender system component which combines two different similarity measures for personalized recommendations, while in Section 4 we illustrate the *Hybrid Recommender System* which utilizes a Bayesian inference algorithm so as to extract a user's preferences before the recommendation process as can be seen in Figure 1. Finally, in Section 5 we illustrate the results of the experiments conducted in real-world data, and in Section 6 we provide a conclusion and future research directions.

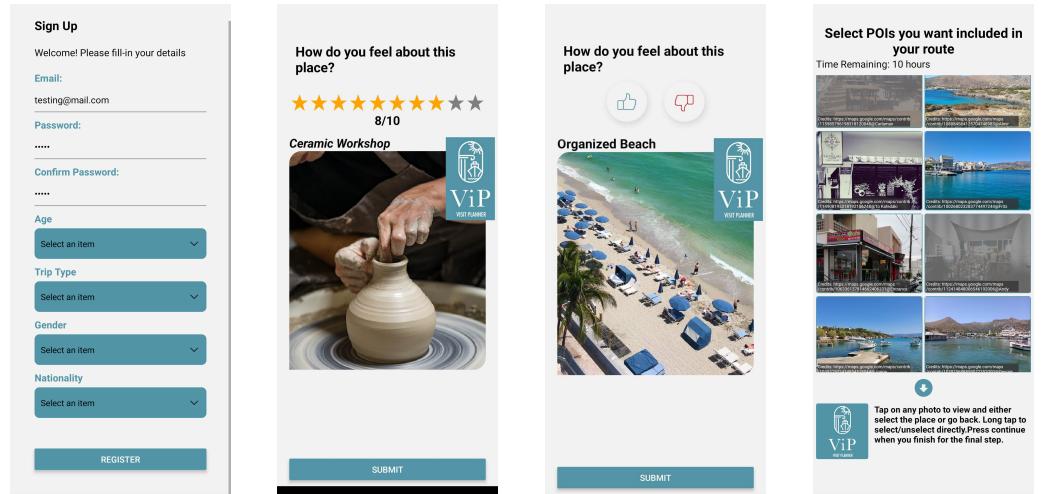


Figure 2. Our *Visit Planner* (*ViP*) real-world RS mobile application: Screenshots.

2. Background and Related Work

Recommender systems are considered to be personalized and non-personalized [4,23]. Personalized recommender systems extrapolate user's preferences to create efficient recommendations based on the users' past interaction with the system [8]. On the contrary, non-personalized recommenders suggest items that are most relevant and popular among all users. For instance the Netflix [24] movie platform recommends the top N items to every user as a list.

Recommender systems may also be classified into several other types: for example, content-based (CB), collaborative filtering (CF), context-aware, Bayesian—while the term "Hybrid" characterizes a recommender that produces recommendations via combining algorithms that belong in several categories. Usually CB and CF are seen as constituting the two main (non-hybrid) approaches [4,5].

In a nutshell, CB RSs exploit information from previous user–system interactions to provide effective recommendations. Particularly, these systems suggest items that are very similar to items the user has liked in previous interactions. CF algorithms, by contrast, analyze user's ratings to calculate the similarity between them. CF recommenders intuitively work under the assumption that, when items are rated by two users in a similar way, they probably share same interests and will provide similar ratings to other items. As mentioned in the introduction, the *cold-start* problem refers to the inability of such algorithms to make recommendations which are personalized and relevant to users' preferences, due to the lack of ratings when dealing with new users or items-to-recommend [19]; this problem is of vital importance in applications related to the tourism domain [20], and thus several ways to tackle cold-start in this domain are proposed in the literature. For instance, Feng et al. [19] propose a ranking model which is a hybrid between a CF ratings-oriented and a Bayesian personalized pairwise ranking-oriented one, while Zheng et al. [20] employ a hybrid CF-based method that refines item opinion reputation and user preferences, by utilizing opinion-mining technology to mine text reviews and

subsequently assess the destinations' preference ranking when matched against preferred features chosen by users via means of an artificial interaction module.

Now, there are numerous travel- or tourism-related RSs, which may be classified into several categories [10]. The majority of these systems suggest POIs that relate to tourist attractions (e.g., monuments, museums or hotels), ideally those that are closely related to the preferences of each visitor while, naturally, in terms of underlying recommendation technology used, a tourism RS may belong to any one of the aforementioned RS classes. We now proceed to briefly review several such systems and algorithms, along with certain recommender approaches that may lie outside the tourism domain but are of interest to our work in this paper.

To begin with, a recommender algorithm, along with a social interactions mechanism, is equipped in a tourist guide application, presented by [25], with the goal of locating undiscovered touristic POIs. An integrated CF system is employed to suggest touristic locations that have been already rated by the users. Lim et al. [26] tackle a tour itinerary planning problem modelled in the context of the well-known orienteering problem, and propose several recommendation algorithm variants that take into account POI popularity to a lesser or greater extent, while assuming user preferences relate to the user's visit durations at POIs of particular categories. The authors of [27] proposed a picture-based recommender technique for proposing tourism sites to a specific individual. Particularly, any set of images is picked by the user and then is imported into computer vision models that generate a profile with respect to the tourist's interests. Sarkar et al. [28] introduced a Crow Search Optimization-based Hybrid Recommendation model capable of generating precise recommendations to travelers through the combination of CB and CF techniques. In their algorithm, undiscovered items are presented to a user based on similar item selection from past interactions. Thus, the similarity between the items is calculated via a combined employment of Jaccard Similarity and Simple Matching Coefficient (SMC) as similarity metrics. In addition, their method is improved as Collaborative Filtering for Java (CF4J) and is enhanced with the Jaccard Similarity metric. They assess experimentally their approach through data provided by the well-known travel-related platform TripAdvisor. Riyani et al. [29] introduced a hybrid RS that works under implicit ratings and semantic similarity in order to provide effective recommendations in a different domain. In greater detail, their suggested method is divided into three filtering components: content-based, collaborative and hybrid, and it makes use of tagging attributes to provide more relevant suggestions on discussion groups. The WordNet lexical database [30] is used to extract the semantic importance of the tags, which are then grouped in a hierarchical framework depending on their semantic relevance.

Now, Bayesian recommenders explicitly model their underlying uncertainty regarding user preferences by efficiently maintaining and exploiting prior distributions over their user models, with the purpose of progressively improving the accuracy of their recommendations. Specifically, Bayesian methods have been proved quite significant in tackling uncertainty in applications with implicit feedback. In [31] researchers derived a generic optimization criterion using a Bayesian analysis of the problem and presented a learning algorithm which is able to provide solutions which satisfy the aforementioned criterion. Sun et al. [32] demonstrated a method using Bayesian Graph Convolutional Neural Networks for modeling the interaction of users with items in implicit recommendations setting. Nevertheless, Bayesian approaches are of course able to deal with and improve recommendations where users share explicit feedback to the system.

For instance, concerning the domain of our interest, the authors of [18] focus on travel personalized recommendations and demonstrate interesting applications by utilizing freely available community-contributed photos. In more detail, they introduce a probabilistic Bayesian framework for mobile recommendations and test it on over ten million images gathered from 19 large cities. Furthermore, Babas et al. [8] propose a Bayesian approach that models *both* the items under recommendation and the user preferences by the same underlying distribution. (In their case, this distribution was the multivariate Gaussian

distribution.) This is what they term as the “*You Are What You Consume*” concept. Their recommendation movie technique exhibits performance results that are comparable to the (at the time) state of the art of a popular CF method for movie recommendations, as shown via an experimental evaluation on data from the MovieLens dataset. It is interesting that this is achieved without the approach having to consult previously obtained or processed data about the user. We adopt certain aspects of their approach during the preference elicitation process of our *Hybrid RS*.

Finally, although providing solutions to a different application domain, a work that bears certain similarities with our, because of the fact that it utilizes semantic similarity measures and hierarchies of items-to-recommend, is that of [33]. The authors, in particular, provide a museum RS for cellphones that merges a CB approach with semantic similarity measures and a semantically enriched CF method to propose relevant museum exhibits to the visitors. Contextual post filtering is implemented to create a personalized tour of the museum depending on the physical environment and location of the visitor. In opposition to our approach, the authors only employ *Wu–Palmer* and *Jaccard* similarity metrics; they do not utilize Bayesian inference. They limit themselves to museum collections since their method is strongly dependent on the usage of specialized ontologies for artworks and cultural heritage, while that paper comes without any kind of evaluation for the approach it proposes.

3. Content-Based Recommender Component

In this section we describe the content-based (CB) component of our Hybrid RS. As mentioned in the introduction, this component makes use of two main sub-components: (i) a *hierarchy* similarity measure-based sub-component and (ii) a hybrid similarity measure-based component which takes as input the output of the aforementioned sub-component and employs a novel non-hierarchy similarity measure (specifically, *WEJS*, the *Weighted Extended Jaccard Similarity*). We hereby refer to the former sub-component as the *Hierarchy Similarity Measure-based Recommendations (Hierarchy SMbR)* algorithm, and to the latter as the *Hybrid Similarity Measure-based Recommendations (Hybrid SMbR)* algorithm.

The *Hierarchy SMbR* calculates the most similar POIs with respect to the user preferences via a well-established hierarchy tree structure. The *Hybrid SMbR* makes use of the aforementioned hierarchy similarity measure and the *WEJS*—which takes into account both the POIs’ features (i.e., *content* of POIs) and the user preferences, as we explain in detail later in this section. Our *CB Recommender* constitutes an algorithmic engine combining the *Hierarchy* and *Hybrid SMbRs* along with two preference elicitation-related parts: (i) the *Show Generic Images*; (ii) the *Build User Profile*, as illustrated in Figure 3. The *POIs Database* stores and also provides all the essential information for the proper operation of our proposed algorithmic engine. The connection of the three components of our implementation with the database is depicted with orange arrows in Figure 3, each of which inserts different input data to the sub-components, as required for their operation.

Now, the construction of the *user profile* is implemented by the combination of the *Show Generic Images* part of the engine that takes as input 90 *generic images* retrieved from the database, and the *Build User Profile* part. In order for the *Hierarchy SMbR* algorithm to work properly, it is important to have an appropriate hierarchy of POIs. To this end, we have constructed a hierarchy of POIs for our application domain, the resort town of Agios Nikolaos, in close collaboration with local stakeholders (e.g., the municipality authorities). A snapshot of the constructed tree hierarchy, containing some of the POIs’ categories we identified, is shown in Figure 4. Our complete hierarchy tree can be found in Appendix A. In our hierarchy, the POIs are inserted in the last layer of the tree structure as leaves, under a layer containing the ninety (90) different POI categories included in the hierarchy. We included 430 POIs in our hierarchy, located in Agios Nikolaos and its vicinity.

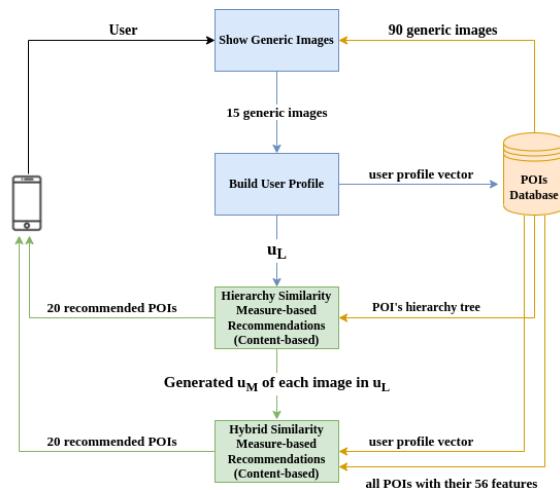


Figure 3. The algorithmic engine of the CB Recommender.

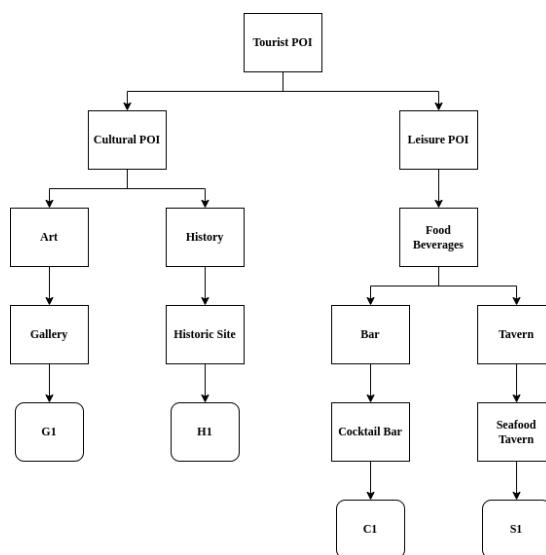


Figure 4. Part of the constructed POIs' hierarchy tree. G1, H1, C1 and S1 are leave-nodes corresponding to specific POIs.

The POIs' database stores our hierarchy and, of course, POIs, and specifies 56 features (each one with a distinct value) for all 430 POIs. The POI's *character* is represented by the first 12 features, with values ranging from 1 to 10, reflecting the extent to which each feature characterizes the POI in question. These 12 "POI character" features were identified as such given the findings of a survey we conducted among real tourists visiting Agios Nikolaos in order to collect data on the preferences of tourists visiting the city. In Appendix B we include the questionnaire we compiled and used in the aforementioned survey. For instance, the POI's *character* could be a combination of cultural and leisure features, e.g., a monastery that offers wine to its visitors. The other 44 features represent the *characteristics* of a POI (e.g., the amenities a POI may offer). In contrast to the *Hierarchy SMbR* which only requires the POIs hierarchy in order to function, the *Hybrid SMbR* exploits all such features of all POIs stored in our database, along with the *user profile vector*. We now proceed to describe all the main parts of the *CB Recommender* algorithmic engine in detail.

3.1. Constructing a User Profile for Content-Based Recommendations

The first process executed by our algorithm is the construction of the user profile. For this purpose, we use two different components, namely the *Show Generic Images* and *Build User Profile* components. Ninety (90) *generic images* (or, *generic POIs*) are also stored

in our database, and are each related to a specific category of POIs. In other words, these *generic POIs* are pictures that illustrate POIs which are not contained to the database, and, importantly, are represented only by the first 12 features which demonstrate the *character* of the *generic POI* depicted in them.

The hierarchy tree we utilize in our algorithm is asymmetrical, i.e., it can have an arbitrary number of children per node, and all branches may have different length. Generic POIs and actual POIs are inserted as leaves in the tree structure, lying under its layer that contains 90 nodes corresponding to the 90 different POI categories. Although every generic image, corresponding as it does to a specific category of POIs, lies on a specific leaf of the hierarchy tree, actual POIs may lie on more than one leaf. As mentioned earlier, in Figure 4 we visualize a snapshot of the POIs' hierarchy structure, generated by real-world data containing specific examples of POIs and categories. Namely, G1 is a gallery and C1 is a cocktail bar, connected to the categories "Gallery" and "Cocktail Bar" respectively. We should highlight that the percentage of POIs and generic images related to "Leisure" is 80%, while those related to "Culture" belong to the other 20%. Those proportions are derived from the responses of actual tourists that participated in the aforementioned survey.

The *Show Generic Images* part of our algorithmic engine depicted in Figure 3 is responsible for selecting randomly and presenting to the user only 15 (The choice of presenting to the user "15" generic images was made since, as [34] indicates, completing a survey with 15 questions requires five to seven minutes only, while pressing a "like" or "dislike" indicator on an image is arguably much faster than responding to a question. In practice, experimentation with our real-world app shows that a real user "likes" or "dislikes" the 15 images presented to her within 45 s) out of 90 *generic images* preserving the proportion of leisure and cultural generic POIs. Subsequently, these 15 *generic images* are imported to the *Build User Profile*. Based on [27], we adopt an elicitation process in which the user either "likes" or "dislikes" each of the aforementioned 15 *generic images* that are presented to her in sequence, resulting in the generation of the *user profile vector*. Notably, the user is allowed to terminate this process at any time (i.e., she does not have to interact with all 15 images).

To build the *user profile*, we act as follows. In accordance with the work of ref. [8], we represent the *user profile* as a *vector* containing the first 12 features of the POIs in our dataset—that is, the features which demonstrate the *character* of a POI. We then populate this vector with values as follows: We first construct the set of all generic images that the user has "liked"—let this set be denoted as u_L . Then, for each one of the 12 "character features", we calculate the average of its values in the u_L set of "liked" generic images and insert this average as the value of the corresponding element in the *user profile vector*.

3.2. Hierarchy Similarity Measure-Based Recommendations

The *Hierarchy SMbR* is a sub-component of the CB recommender algorithm which is capable of providing recommendations via a hierarchy similarity measure called *XWP*. This metric is based on a similarity measure introduced in [16], which is an extended version of *Wu–Palmer (WP)* [35]. More specifically, the semantic similarity between two concepts X, Y , by employing *WP*, is described as:

$$WP(X, Y) = \frac{2N}{E_1 + E_2} \quad \text{with } X \neq Y \quad (1)$$

where E_1, E_2 represent the number of edges between the *root node* and the ontologies X, Y respectively, while N is the number of edges between the *root node* and the *Least Common Ancestor (LCA)* of X and Y . When the *WP* similarity score is close to 0, ontologies X, Y are very distant, while when it approaches 1, X and Y are quite similar.

The extended version of *WP*, or *XWP* as we denote it, computes the semantic relation between the ontologies X, Y and is defined as:

$$XWP(X, Y) = WP(X, Y) \cdot e^{\frac{-L\lambda}{H}} = \frac{2N \cdot e^{\frac{-L\lambda}{H}}}{E_1 + E_2} \quad (2)$$

where H is the height of the hierarchy structure and L represents the minimum distance between ontologies X and Y . Furthermore, the λ factor is zero when the compared ontologies' LCA is in their neighborhood, otherwise it equals one. In our instantiation of the XWP measure, we define the neighborhood of a particular node to be the set of all other nodes which have a distance less than or equal to 2 from it.

The left part of Figure 5 depicts an overview of the *Hierarchy SMbR*-based recommendation process.

Specifically, the u_L vector of “liked” images is given as an input to the *Hierarchy SMbR* sub-component and for every “liked” generic image contained in this set we compute the XWP similarity distance between it and each POI in our database. Now, the M most relevant POIs to each generic image i according to XWP are stored in a set, denoted as u_M^i , alongside their similarity scores. For instance, as Figure 5 illustrates, if the user has “liked” N generic images during the elicitation process, our system will create one set for every “liked” image, i.e., $u_M^1, u_M^2, \dots, u_M^N$. The POIs included in all u_M sets are grouped together, and their corresponding XWP similarity scores are transformed into a probability distribution by a normalization process—i.e., the *Normalization Process* part of the *Hierarchy SMbR* component (illustrated in blue) in Figure 5. Finally, if the standalone operation of this sub-module is desired, we sample out 20 POIs in order to be presented to the user.

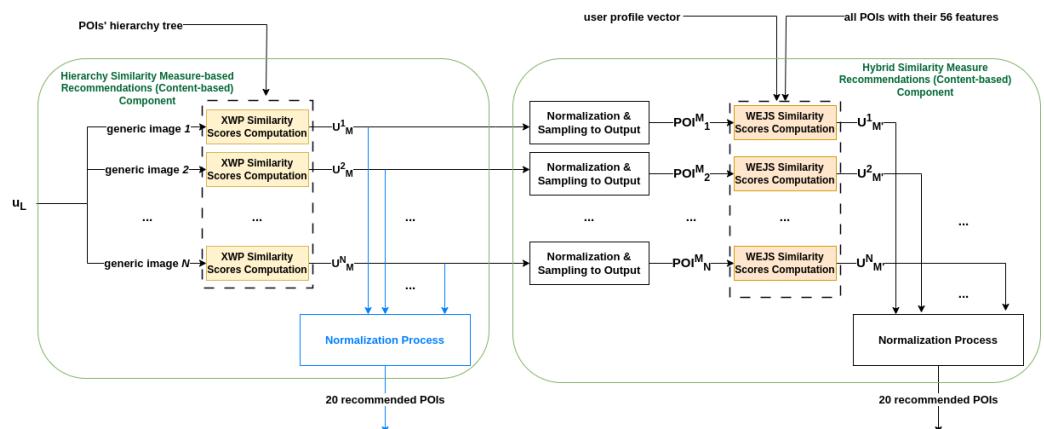


Figure 5. Overview of our Content-based (CB) Recommender.

3.3. Hybrid Similarity Measure-Based Recommendations

However, the *Hierarchy SMbR* sub-component need not operate alone, but can be extended to our (“hybrid”) CB recommender as follows.

We now add a sub-component that performs *Hybrid similarity measure-based recommendations* (right part of Figure 5), and which incorporates a non-hierarchy similarity measure, the Weighted Extended Jaccard Similarity (WEJS) [22], combining it with the (hierarchy-based) XWP. In detail, as illustrated in Figure 5, the *Hybrid SMbR* sub-component takes three different inputs: (i) the u_M sets produced by *Hierarchy SMbR* (we remind the reader that each such set corresponds to a generic image in u_L); (ii) the *user profile vector* (i.e., the 12 feature-based vector generated by the *Build User Profile* component); and (iii) the POIs included in the database alongside their features. Our system transforms the XWP similarity outcomes into probability distributions for each u_M produced by the hierarchy similarity measure via the *Normalization & Sampling to Output* component. Specifically, after this transformation, each XWP similarity score of a POI in a specific u_M set is mapped to a value in the range of $[0,1]$, while the summation of all such values for POIs in the u_M in question equals 1. Given these probabilities, we sample out of each u_M (corresponding to an image in u_L), one POI for computing its WEJS similarity score (via the Equation (4) formula explained below in detail). As Figure 5 indicates, we denote the drawn POI of u_M^i set as POI_M^i , where $i \in \{1, \dots, N\}$. After the employment of WEJS similarity score, for each POI_M^i , a new set, denoted as u_M^i' , is generated, and it contains at most M' similar POIs

along with their corresponding *WEJS* similarity scores. Following this, all generated sets $u_{M'}^i, i \in \{1, \dots, N\}$, are concatenated, and their *WEJS* similarity scores are converted into probabilities. Finally the system samples out 20 POIs, and recommends them to the user.

Now, *WEJS* is a novel metric computing the relevance of sets of ontologies inspired by the well-known *Jaccard Similarity (JS)* one, which is calculated by this formula:

$$JS(X, Y) = \frac{|X \cap Y|}{|X \cup Y|}, \quad (3)$$

where $|X \cap Y|$ represents the cardinality of the intersection of sets X and Y , while $|X \cup Y|$ constitutes the size of the union of the aforementioned sets. JS is zero when the compared sets are disjoint, while it is equal to one when the sets are identical.

On the other hand, *WEJS* is computed by the following expression:

$$WEJS(X, Y) = \frac{\sum_{i \in X \cap Y} \alpha \cdot w_i}{\sum_{j \in X \cup Y} w_j}, \quad \alpha = \begin{cases} 1, & \text{if } v_i^X = v_i^Y \\ 0, & \text{otherwise.} \end{cases} \quad \text{with } i \in X \cap Y \quad (4)$$

where X and Y in our application domain correspond to two distinct POIs—i.e., to the sets of these POIs' feature values. Moreover, v_i^X and v_i^Y represent the value of the i^{th} feature of X 's and Y 's intersection; thus, α is 1 if they are identical, otherwise it is 0. w_i (respectively w_j) is given by the output of Algorithm 1, and corresponds to the weight of the i^{th} (j^{th}) feature of the intersection (union) of POIs X, Y . As such, *WEJS* measure exploits the generated weights of all POI's last 44 features (i.e., *characteristics*) with the purpose of comparing the intersection members (i.e., features) of two POIs, and it utilizes only those with equal values. Intuitively, two items would be considered by *WEJS* highly similar, and also of high recommendation value to a user, if they share the exact same characteristics (i.e., share features with the exact same values), and these characteristics are deemed important (i.e., they are highly “weighted” by the user) [22].

As mentioned, Algorithm 1 presents the method of generating the weights vector with respect to the user preferences for the POI's *characteristics*, or, as we term it, the *weights user profile vector*. As seen in Algorithm 1, the cosine similarity distance metric [36] is employed between the twelve features of the *user profile vector* and each actual POI's twelve *character* features. The most similar POIs are selected according to a *cosine threshold*. This threshold was empirically set to 0.70 during in our experiments. Then, the total appearances of their last 44 features with non-empty values are counted. Finally, these features' counters are normalized into weights comprising the generated $weightsProfile_u$ of the user, to be used in the *WEJS* calculation.

Algorithm 1 Weights generation of the user profile

```

1:  $profile_u \leftarrow$  user profile vector
2:  $POIs \leftarrow$  all POIs with their 56 features
3:  $counters \leftarrow$  count vector of length 44 initialized to zero
4: for each  $POI$  in  $POIs$  do
5:    $character_{POI} \leftarrow$  first 12 features of  $POI$ 
6:    $characteristics_{POI} \leftarrow$  last 44 features of  $POI$ 
7:   if  $cosine_{sim}(profile_u, character_{POI}) \geq 0.70$  then
8:     for each  $f$  in  $characteristics_{POI}$  do
9:       if  $v_f \neq \emptyset$  then
10:         $counters_f \leftarrow counters_f + 1$ 
11:  $weightsProfile_u \leftarrow normalize(counters)$ 

```

4. A Hybrid Recommender System

Here, we describe our *Hybrid RS* which combines the aforementioned *CB Recommender* with a *Bayesian Inference* component used for user profile elicitation. During that process,

we adopt the *You Are What You Consume* approach of [8]: that is, we model the POIs of the travel destination (i.e., items) and the users, using multivariate normal distributions over ranges of values as a common representation, describing the degree that each feature describes a specific user or item. Figure 6 depicts the overall architecture of the proposed system. Notice how the elicitation process of Section 3.1 (or, the blue boxes in Figure 3) is now replaced with a process of *Bayesian Inference*, the functionality of which we explicate immediately below.

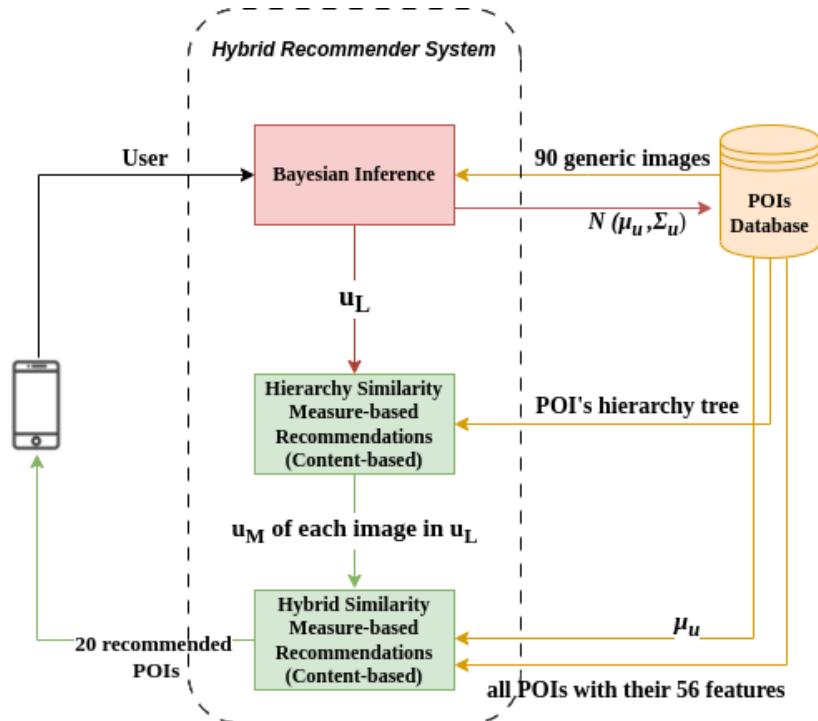


Figure 6. The algorithmic engine of the proposed Hybrid RS.

4.1. Bayesian Inference of the User Profile

Initially, when a user enters our *Hybrid RS* for the first time, the preference elicitation process—i.e., the *Bayesian Inference* component in Figure 6)—initiates. We chose to adopt an iterative picture-based elicitation process that presents to the user a set of n generic images, each of which illustrates a specific type of POI, in order to derive information regarding her interests. Subsequently, the user selects the image which is most relevant to her interests and provides a rating for each image on a 10-level Likert scale, where 10 represents the situation where the image matches perfectly with her preferences. Note that a detailed description of this process can be found in [11]. After each iteration, our system exploits the provided rating of the selected generic image and draws a specific number of samples from its distribution. The number of samples ranges from 40 to 500 depending on the user’s rating. Intuitively, the user’s rating and the logistic function [37] are utilized with the aim of computing the precise number of samples

In more detail, a high rating provided by the user for a generic image represents high similarity between the distributions of the image and the user, while a small rating means that this image does not match completely with her preferences. As such, more samples should be drawn in the case of the high rating resulting to the construction of a better model based on the user’s interests. Subsequently, once our algorithm has computed the exact number of samples that should be drawn, it performs a Bayesian Inference procedure [8,11] in order to update its belief regarding the interests of the user. In particular, we chose to employ the Normal-Inverse Wishart (NIW) distribution which is a multivariate four-parameter family of continuous probability distributions, while it retains the property of *conjugacy* of a multivariate Gaussian distribution where the mean and covariance matrix

are unknown. Generally, a closed form for the computation of the posterior distribution is provided by the usage of conjugate priors, resulting in a computationally efficient Bayesian updating procedure. Formally, the update procedure of the prior hyperparameters— κ_0 , μ_0 (the mean vector), v_0 (degrees of freedom) and Ψ_0 (the precision matrix)—is performed by drawing samples directly from the observed data to obtain the posterior ones, as follows:

$$\mu_n = \frac{\kappa_0}{\kappa_0 + n} \cdot \mu_0 + \frac{n}{\kappa_0 + n} \cdot \bar{x} \quad (5)$$

$$\kappa_n = \kappa_0 + n \quad (6)$$

$$v_n = v_0 + n \quad (7)$$

$$\Psi_n = \Psi_0 + S + \frac{\kappa_0 \cdot n}{\kappa_0 + n} \cdot (\bar{x} - \mu_0) \cdot (\bar{x} - \mu_0)^T \quad (8)$$

$$S = \sum_{i=1}^n (x_i - \bar{x}) \cdot (x_i - \bar{x})^T \quad (9)$$

where n is the number of samples, x_i are the drawn samples provided by the data, \bar{x} represents the sample mean and S is the scatter matrix.

$$\Sigma \sim IW(\Psi_n, v_n) \quad (10)$$

$$\mu \mid \Sigma \sim N(\mu_n, \Sigma / \kappa_n) \quad (11)$$

Thus, our algorithm can generate a normal distribution representing the user preferences at any given time: after m iterations our system has constructed a multivariate normal distribution that represents the preferences of user u , denoted as $\mathcal{N}(\mu_u, \Sigma_u)$, where μ_u and Σ_u are the mean vector and the covariance matrix of the generated distribution, respectively.

Additionally, our system stores the selected image from each iteration in order to exploit such information in the *CB recommendations* stage of the hybrid approach. As such, during the elicitation process, we create a set, denoted as u_L , which contains the images that the user selected in each iteration.

Thus, the output of our Bayesian elicitation process is: (i) a multivariate Gaussian distribution, $\mathcal{N}(\mu_u, \Sigma_u)$, that describes the preferences of user u , where μ_u is employed so as to compute the weights in *WEJS* (i.e., μ_u replaces the *user profile vector* in Algorithm 1); and (ii) the set u_L that contains the generic images that u has selected as most preferred during the elicitation process.

4.2. Content-Based Recommendations

Subsequently, our proposed system proceeds with our *content-based (CB) recommendation* technique. In particular, our system employs the *hybrid* similarity measure (*Hybrid SMbR*) discussed earlier, in order to produce the final set of personalized recommendations. We remind the reader that, via this hybrid measure, our algorithmic engine combines *XWP* similarity, (i.e., the hierarchy similarity measure employed in the *Hierarchy SMbR* sub-component in Figure 6—see Section 3.2), and our *Weighted Extended Jaccard Similarity (WEJS)* index (i.e., a non-hierarchy similarity measure employed in the *Hybrid SMbR* component in Figure 6—see Section 3.3), to generate recommendations that best match the user preferences.

Figure 7 depicts the *CB recommender component* of the *Hybrid RS*. Notice that this figure is almost identical to Figure 5, with two changes: First, the *Hierarchy SMbR* sub-component is now not allowed to produce final recommendations (the blue *Normalization Process* and its output are now removed), but only provides input to the *Hybrid SMbR* sub-component.

This is since our experiments in Section 5 point to the *Hybrid SMbR* being a better choice (than solely *Hierarchy SMbR*) for a *CB recommender* component of our final *Hybrid RS*. The second change is that μ_u replaces the *user profile vector* as input in the *WEJS* metric, since our (hybrid) *CB recommender* is now combined with the Bayesian elicitation module that produces a user profile in the form of a $\mathcal{N}(\mu_u, \Sigma_u)$ multivariate Gaussian distribution.

The implementation of the overall *Hybrid recommendations process* is summarized in Algorithm 2 below. The algorithm begins with the *Bayesian Inference* process, which produces the set u_L and the distribution $\mathcal{N}(\mu_u, \Sigma_u)$ describing the derived user profile. It proceeds with the generation of u_M^X sets by employing *XWP similarity* between the u_L set and the POIs in our dataset (see line 7 in Algorithm 2). Subsequently, one POI is sampled out from each u_M^X set via the *Normalization & Sampling to Output* component, and is provided as input to the *WEJS* measure along with the μ_u (see line 12 in Algorithm 2). Finally, for each selected POI a (new) $u_{M'}^S$ set is created, and all such sets are concatenated (see line 13 in Algorithm 2) in order to derive the final recommendations.

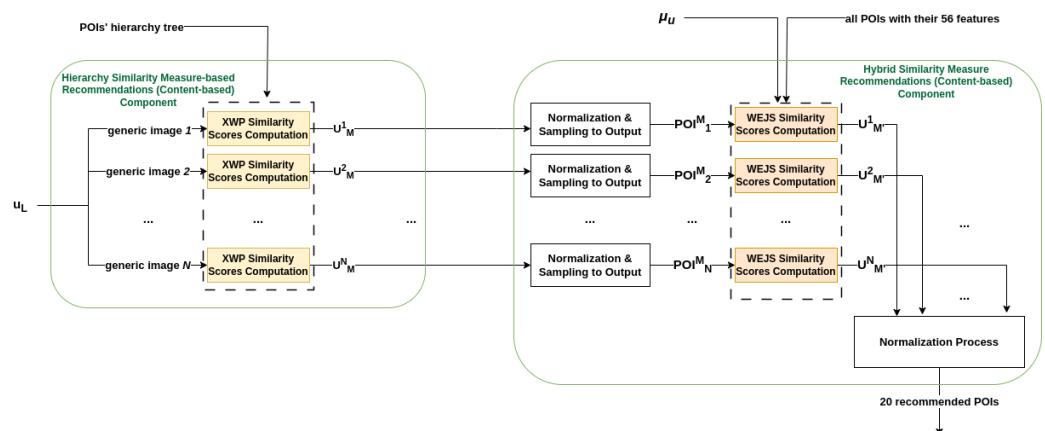


Figure 7. Overview of the CB recommender part of the Hybrid RS.

Algorithm 2 Hybrid Recommender Algorithm

```

1:  $u_L, \mathcal{N}(\mu_u, \Sigma_u) \leftarrow$  Bayesian Inference
2:  $POIs \leftarrow$  all POIs with their 56 features
3: for each generic image in  $u_L$  do
4:    $X \leftarrow$  generic image
5:   for each POI in POIs do
6:      $Y \leftarrow$  POI
7:      $u_M^X \leftarrow u_M^X \cup is\_XWP\_similar? (X, Y, POIs' Hierarchy)$ 
8:     Single POI  $\leftarrow$  Normalization_and_Sampling_to_Output ( $u_M^X$ )
9:      $S \leftarrow$  Single POI
10:    for each POI in POIs do
11:       $Y \leftarrow$  POI
12:       $u_{M'}^S \leftarrow u_{M'}^S \cup is\_WEJS\_similar? (S, Y, \mu_u)$ 
13:    Recommendations  $\leftarrow$  Recommendations  $\cup u_{M'}^S$ 
14: 20 Recommended POIs  $\leftarrow$  Normalization (Recommendations)

```

5. Experimental Evaluation and Results

In this section we assess the performance of our algorithms on real-world data comprising 430 points of interest from the city of Agios Nikolaos by using a lab computer with AMD Ryzen 7 3700x 8-core processor x 16 CPU with a GeForce RTX 3080 GPU. We collected information related to *users' preferences* from questionnaires filled out by 150 actual short-term visitors and used these data to generate a set of 600 synthetic users. We categorized the synthetic users to six different age classes (100 tourists per class), namely 17–25, 26–35, 36–45, 46–55, 56–67 and 67+.

In order to generate the synthetic users' profiles, we proceeded as follows. We first computed an average profile for each age class, containing the average values of the 12 features representing user preferences, as listed in the filled-in questionnaires of the real tourists belonging to that particular class. Specifically, these 12 features were used: *Family friendly facilities and activities, Luxury Accommodation, Affordable Prices, Culture, Shopping Local Products, Sun and Sea, Rural Tourism, Sports and Adventure, General Shopping, Nightlife, Cuisine and Gastronomy, Archaeology or History*. (We remind the reader that these 12 features correspond to the 12 "POI character" features, in accordance to the "You Are What You Consume" approach.) Subsequently, for each class, we sampled 100 random user profiles from a multivariate normal distribution, generated using the mean vector from the average "real tourist" profile mentioned above; its covariance matrix is diagonal with all its eigenvalues equal to 2.

We assess the *content-based* and *Hybrid* recommender (described in Sections 3 and 4) separately in the following subsections, and we compare their results. In our experimental setting each (synthetic) user interacts with the recommenders 30 times (each time without storing any information about the model for the next run of the system). Thus the set of generic images provided to her is not necessarily always the same. As explained in the previous sections, the elicitation process differs between the two CB (see Section 3.1) and Hybrid (see Section 4.1) recommenders, both in the number of generic images presented to the user and their functionality.

For each user, we utilize k -means clustering [38] to classify all POIs to two distinct categories, specifically, those that are close to her profile, termed the *relevant POIs*, while the complementary set is that of *irrelevant POIs*. Since we are interested primarily in the intersection of POIs which are relevant to those recommended by each of our algorithms, we denote as Rec_r the average cardinality of this set of POIs over all 100 users per age group, and over 30 runs per user.

We evaluate the performance of our algorithms by employing the well-known *precision* and *recall* measures of recommendations' quality [39]. Notably, precision is the quotient of the number of recommended relevant POIs over the number of all (relevant and irrelevant) recommended POIs; the latter, in our case, is always 20. Recall is the quotient of the number of recommended relevant POIs over the number of all (recommended and non-recommended) relevant POIs.

Now, let us provide an intuitive interpretation of precision and recall. Recall decreases when the non-recommended relevant POIs increase, while the precision is greater when the algorithm recommends fewer irrelevant POIs. Therefore, we can utilize recall in order to provide an assessment tool for users who evaluate the performance of our application based on whether they did not visit a relevant POI. On the other hand, we employ precision to model the evaluation of our algorithms from users who are interested in not visiting any irrelevant POI. Furthermore, we highlight that the number of recommended relevant POIs cannot exceed 20, however the relevant POIs are far more than this number; thus, the value of recall will always be much lower than 1. For instance, if the number of relevant POIs is 272, recall cannot exceed 0.073. Hence, recall in our results should always be assessed based on its corresponding maximum value, denoted below as *Recall Upper Bound* (or *Recall UB* for short). We focus mainly on the value of *precision* for reasons of clarity, and because we (reasonably) assume that the users' "assessment-profile" is closer to that of precision rather than recall, *intuitively, short-term visitors cannot possibly explore every relevant POI with respect to their interests, but they tend to be frustrated when they spend time on irrelevant POIs*.

5.1. Evaluation of Our Content-Based Recommendation Algorithms

We first conduct a series of experiments in order to evaluate the performance of the semantic similarity-based algorithms that are contained in the CB component of our system—i.e., the methods introduced in Sections 3.2 and 3.3 of this paper. Since our experiments involve synthetic users, our CB recommendation process requires the identification of images a synthetic user "likes". This is carried out by comparing the *cosine similarity*

among the values of the (twelve) features of the “*actual*” (synthetic) user profile and the values of the corresponding features representing the (fifteen) “generic” images displayed to the user by the elicitation component (termed “Show Generic Images” in Figure 3) of the CB recommendation process. In case the cosine similarity between an image and the profile exceeds a threshold, set to 0.80, the image is considered to be “selected” (or “liked”) by the user; otherwise it is not. The “liked” images make up the u_L set of images fed into the *Hierarchy SMbR*, while the *Build User Profile* sub-component (cf. Figure 3) also constructs a *user profile* that the (hybrid) CB recommender can use for its recommendations. We now proceed with the comparison between the performance of the *hierarchy SMbR* algorithm (i.e., the output of the *Normalization Process* in blue color in Figure 5) and the *hybrid SMbR* (i.e., the output of the *Normalization Process* in black color in Figure 5).

In Tables 1 and 2 we illustrate the aforementioned measures of quality (i.e., precision and recall) of our application, together with the number of relevant POIs, Rec_r , and *Recall UB* for each age group. (We remind the reader that *Recall UB* (i.e., Recall Upper Bound) stands for the maximum possible value of Recall that could be attained by an algorithm capable of “optimally” eliciting and exploiting users’ preferences in this setting, i.e., no algorithm could have performed better than this value.) The results of Table 1 concern the *Hierarchy SMbR*, while those of Table 2 correspond to the *Hybrid SMbR* of the CB recommender. We observe that the results (i.e., the values of *Precision*, and the respective values of *Recall* compared to *Recall UB*) between the two recommendation processes are quite satisfactory, as *Precision* is regularly higher than or close to 80%, while *Recall* reaches values close to or higher than 80% of the respective *Recall UB* ones. In general, the two CB methods appear to be comparable to each other: *Hierarchy SMbR* achieves a better performance for younger users, while *Hybrid SMbR* attains greater values of precision and recall for the age groups 56–67 and 67+.

Table 1. *Hierarchy SMbR* evaluation results.

<i>Hierarchy Similarity Measure-Based Recommendations</i>					
Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.782	0.057	0.073	272.4	15.6
26–35	0.817	0.059	0.072	275.3	16.2
36–45	0.781	0.057	0.072	274.2	15.6
46–55	0.821	0.060	0.074	269.3	16.4
56–67	0.862	0.063	0.073	272.9	17.2
67+	0.684	0.055	0.080	248.9	13.7

Table 2. *Hybrid SMbR* evaluation outcomes.

<i>Hybrid Similarity Measure-Based Recommendations</i>					
Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.762	0.055	0.073	272.5	15.2
26–35	0.800	0.059	0.072	275.4	16.0
36–45	0.761	0.056	0.072	274.3	15.3
46–55	0.819	0.059	0.074	269.3	16.3
56–67	0.879	0.064	0.073	273.0	17.5
67+	0.721	0.058	0.080	249.2	14.4

In addition, Table 3 illustrates the average precision of the algorithms across all age groups. We can observe that the *Hybrid SMbR* has an average precision value that is slightly greater than the one of the *Hierarchy SMbR*. Of course, this small difference in precision alone would not be enough in order to pick the *Hybrid SMbR* as the preferable CB recommender system between the two. However, another drawback of the *Hierarchy SMbR* is that the u_M set is always the same for all users who “liked” a particular generic image, as such the generated recommendations would be quite similar for these users. Moreover,

the method does not take into account a user profile. These characteristics clearly do not contribute to diverse or personalized recommendations.

Table 3. Average precision values for the CB recommendation methods.

Hierarchy SMbR	Hybrid SMbR
0.7911	0.7913

On the other hand, the *WEJS* measure used by the *Hybrid SMbR* module combines information about the user's preferences and the content of the POIs, by taking into account the *user profile vector* and all 56 features of the POIS. As such, we expect *Hybrid SMbR* to result in recommendations that are both more personalized and more diverse. We explicate the latter point and the meaning of the term "diversity", by providing an example: when the system recommends a monastery which produces its own wine, it is probable to also recommend a liqueur store that sells the monastery's wine, even though liqueur stores are not in the vicinity of monasteries in the hierarchy tree, thus *Hierarchy SMbR* could not have possibly captured this connection. (This point is also supported by the experiments in [22] evaluating the *WEJS* measure.) Moreover, the *Hybrid SMbR* is also able to exploit information about the user profile captured via the elicitation process. For those reasons, we pick the *Hybrid SMbR* to be the *CB recommender* method of choice for our overall *Hybrid RS*, which we now proceed to evaluate.

5.2. Evaluation of the Complete Hybrid Recommender System

In this section we evaluate the complete *Hybrid* recommender algorithm. As explained earlier, we now adopt the iterative elicitation process introduced in Section 4.1, which results in a multivariate Gaussian distribution, $\mathcal{N}(\mu_u, \Sigma_u)$, that describes the preferences of user u . In detail, we run a series of user–system interactions with varying "slates". A slate in our case corresponds to an implicit "query" presented to the user, comprising of a number n of images for the user to select a preferred one from. We experiment with varying numbers of slates $m = \{4, 5, 6\}$ posed to the user, and with varying numbers $n = \{4, 5, 6\}$ of generic images presented to the user in a slate ("query"). The testing of various such m, n combinations of *elicitation slates, images shown per slate* pairs, aims to assess the trade-off between the information offered to the algorithms (expected, naturally, to improve their performance), and the frustration caused to users when the time required from them to spend "marking" (generic) images for elicitation purposes increases [40]. Thus, as also explained in Section 4.1, during each $\langle m, n \rangle$ elicitation instance, the user selects the generic images that are more appealing to her, given her interests. The "selected" images are stored in the u_L set, and used alongside μ_u to recommend POIs to the user (as explained in Figure 7 and Algorithm 2).

In Tables 4–12 we present the average values of precision and recall, alongside with the (average) number of relevant POIs, Rec_r (the average number of relevant POIs among the 20 recommended) and $Recall\ UB$ for all age groups, each of which comprises 100 users who enter the system 30 times. Furthermore, we illustrate the average precision values for every $\langle m, n \rangle$ combination in Table 13.

As seen in Table 13, our *Hybrid RS* outperforms the aforementioned *Cb* algorithms regarding the average values of *Precision* for all pairs of $\langle m, n \rangle$, and also almost always outperforms them in terms of *Recall* (the latter's values are regularly about 85% of those of *Recall UB*, i.e., our method regularly returns about 17 recommendations that are relevant to the user profile, out of the 20 relevant recommendations that it could have possibly returned). However, slightly higher precision outcomes are observed for some age groups (namely 56–67, 67+) for the *Hybrid SMbR* CB recommender (see Table 2). Overall, it is clear that the *Hybrid RS* has a much more stable performance than the *CB* recommenders, achieving outcomes that are consistently higher than 0.84 for all pairs (apart from the 67+ age group), and consistently including about 17 POIs that are relevant to the user interests, in its 20 returned recommendations.

According to Table 13 our *Hybrid* recommender achieves the best results for its variant using values of $m = 4$, $n = 6$ during elicitation—i.e., when the user is presented four times with six images each time. Now, for larger values of combinations m, n , the cardinality of the set u_L is larger ($|u_L| \geq m$). This creates the possibility for more irrelevant POIs being contained in the sets u_M and $u_{M'}$, and as a result they may be presented in the final recommendations to the user—naturally, when the proportion of the irrelevant POIs is high, the precision results decrease. We can see a possible manifestation of this behaviour in the precision outcome of the $m = 6$, $n = 6$ pair, which is lower than the respective $m = 4$, $n = 6$, even though more information is potentially elicited from the user in the $m = 6$, $n = 6$ case. That information probably results to a “blurring” of the user model.

Hence, since it is better than all other combinations for average precision, as listed in Table 13, combined with the fact that being presented with fewer slates ($m = 4$ instead of 5 or 6) results in a “lighter”, less tiresome elicitation process for the user, we consider the variant of our *Hybrid RS* that uses the $\langle m = 4 \text{ elicitation slates}, n = 6 \text{ images shown per slate} \rangle$ pair as the one to incorporate into our final system.

Table 4. Hybrid RS performance; ($m = 4, n = 4$).

Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.833	0.060	0.073	272.4	16.6
26–35	0.869	0.063	0.072	275.3	17.3
36–45	0.849	0.062	0.073	273.7	16.9
46–55	0.844	0.062	0.074	270.1	16.8
56–67	0.840	0.061	0.073	272.6	16.8
67+	0.705	0.056	0.080	249.4	14.1

Table 5. Hybrid RS performance; ($m = 5, n = 5$).

Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.862	0.063	0.073	272.6	17.2
26–35	0.886	0.064	0.072	275.4	17.7
36–45	0.868	0.063	0.072	274.0	17.3
46–55	0.855	0.062	0.074	269.6	17.1
56–67	0.858	0.062	0.073	272.8	17.1
67+	0.706	0.056	0.080	249.6	14.1

Table 6. Hybrid RS performance; ($m = 6, n = 4$).

Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.853	0.062	0.073	272.7	17.0
26–35	0.886	0.064	0.072	275.2	17.7
36–45	0.869	0.063	0.072	274.3	17.3
46–55	0.861	0.063	0.074	269.6	17.2
56–67	0.854	0.062	0.073	272.8	17.0
67+	0.714	0.056	0.080	249.7	14.2

Table 7. Hybrid RS performance; ($m = 4, n = 6$).

Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.858	0.062	0.073	272.7	17.1
26–35	0.893	0.064	0.072	275.5	17.8
36–45	0.864	0.063	0.073	273.9	17.2
46–55	0.868	0.063	0.073	270.5	17.3
56–67	0.866	0.063	0.073	272.8	17.3
67+	0.725	0.058	0.080	249.5	14.5

Table 8. Hybrid RS performance; ($m = 5, n = 4$).

Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.851	0.062	0.073	273.0	17.0
26–35	0.880	0.063	0.072	275.5	17.6
36–45	0.859	0.062	0.073	273.9	17.1
46–55	0.853	0.062	0.074	269.9	17.0
56–67	0.846	0.061	0.073	273.0	16.9
67+	0.702	0.056	0.080	249.0	14.0

Table 9. Hybrid RS performance; ($m = 4, n = 5$).

Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.843	0.061	0.073	272.6	16.8
26–35	0.879	0.063	0.072	275.7	17.5
36–45	0.861	0.062	0.072	274.1	17.2
46–55	0.854	0.063	0.074	269.2	17.0
56–67	0.855	0.062	0.073	272.8	17.1
67+	0.716	0.057	0.080	249.4	14.3

Table 10. Hybrid RS performance; ($m = 6, n = 5$).

Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.851	0.062	0.073	272.2	17.0
26–35	0.872	0.063	0.072	275.2	17.4
36–45	0.849	0.061	0.072	274.1	16.9
46–55	0.841	0.061	0.074	270.0	16.8
56–67	0.852	0.062	0.073	272.7	17.0
67+	0.717	0.058	0.080	248.9	14.3

Table 11. Hybrid RS performance; ($m = 5, n = 6$).

Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.864	0.063	0.073	272.9	17.2
26–35	0.886	0.064	0.072	275.7	17.7
36–45	0.868	0.062	0.072	274.4	17.3
46–55	0.845	0.061	0.074	270.1	16.9
56–67	0.855	0.062	0.073	273.0	17.1
67+	0.708	0.059	0.080	249.0	14.6

Table 12. Hybrid RS performance; ($m = 6, n = 6$).

Age Group	Precision	Recall	Recall UB	Relevant Items	Rec_r
17–25	0.855	0.062	0.073	272.3	17.1
26–35	0.876	0.063	0.072	275.2	17.5
36–45	0.853	0.062	0.073	273.7	17.0
46–55	0.840	0.061	0.073	270.3	16.8
56–67	0.841	0.061	0.073	272.8	16.8
67+	0.710	0.057	0.080	249.2	14.2

Table 13. Avg. precision per $\langle m, n \rangle$ pair.

	$n = 4$	$n = 5$	$n = 6$
$m = 4$	0.823	0.834	0.846
$m = 5$	0.831	0.839	0.834
$m = 6$	0.839	0.830	0.829

We remind the reader that we primarily focus on precision values, since it can measure the frustration of users spending time on irrelevant POIs (as discussed in the beginning of Section 5). Interestingly, we note that the performance of our methods in terms of precision is quite satisfactory (as precision almost always exceeds 0.8), also when contrasted with the performance of other recommendation algorithms used in the tourism domain: for instance, the approaches of [20,26] report precision values under 0.8. In addition, our algorithms effectively circumvent the cold-start problem, which is key for the quality of the recommendations, especially in the setting of our mobile application (i.e., short-term tourist visits).

6. Conclusions and Future Work

In this work a novel *Hybrid Recommender System* is introduced for the tourist domain, combining a Bayesian preference elicitation technique and a content-based recommendation process. The experimental assessment of the proposed *Hybrid Recommender System*, as well as the content-based recommendation techniques is performed on a real-world dataset, constructed during the development of a real-world travel planning mobile application, and incorporated knowledge derived from a survey involving actual tourists. The content-based recommender component of our system combines a hierarchy similarity measure with a novel non-hierarchical one. Our experimental results verify the effective performance of several variants of our proposed *Hybrid RS*, indicating that—compared to the exclusively content-based techniques—they return many more POIs relevant to the inferred user profile. In addition, when comparing the performance of the various *Hybrid RS* variants against each other, we are able to come up with a “winner”: the most preferable *Hybrid RS* variant is the one using an $\langle m = 4 \text{ elicitation slates}, n = 6 \text{ shown images per slate} \rangle$ pair as input to its *Bayesian Inference* component.

Now, despite the fact that our experimental findings are encouraging, they were performed with only synthetic users and, while we concentrated on a specific real-world dataset linked to our recommendation algorithm for the location of interest, it would be beneficial to evaluate our algorithms with various other datasets. In ongoing and future work, we aim to conduct a large-scale evaluation of our system via utilizing feedback provided by real users of our (already available online) mobile application, following all required policies and provisions for protecting users’ privacy. Since different versions of our real-world application incorporate several recommender algorithms (additional to those presented in this paper), such experiments will also provide us with the opportunity to compare these methods against each other, and to come up with modifications, algorithmic improvements, and even novel hybrid solutions.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, G.C., I.Z., E.S. and H.P.; Funding acquisition, G.C. and H.P.; Investigation, G.C., I.Z. and E.S.; Methodology, G.C., I.Z., E.S., C.P. and H.P.; Software, I.Z., M.K. and E.S.; Validation, G.C., I.Z., M.K. and E.S.; Supervision, G.C. and H.P.; Data Curation, I.Z. and E.S.; Writing—original draft, G.C., I.Z., M.K. and E.S.; Writing—review & editing, G.C., I.Z., M.K., E.S., C.P. and H.P. All authors reviewed the results and approved the final version of the manuscript.

Funding: 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).

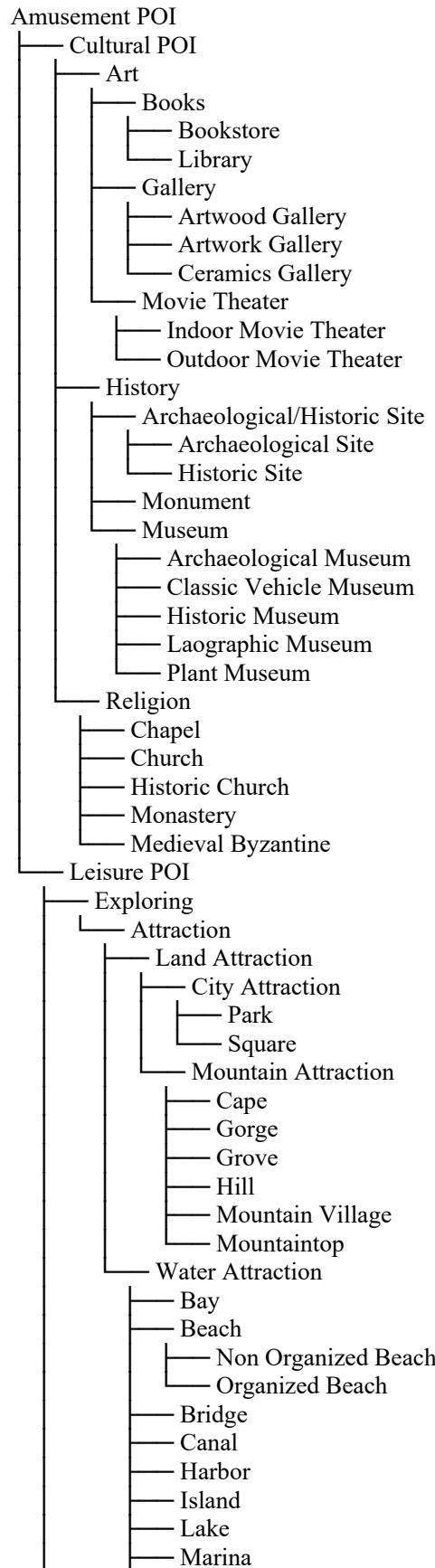
Institutional Review Board Statement: Not applicable.

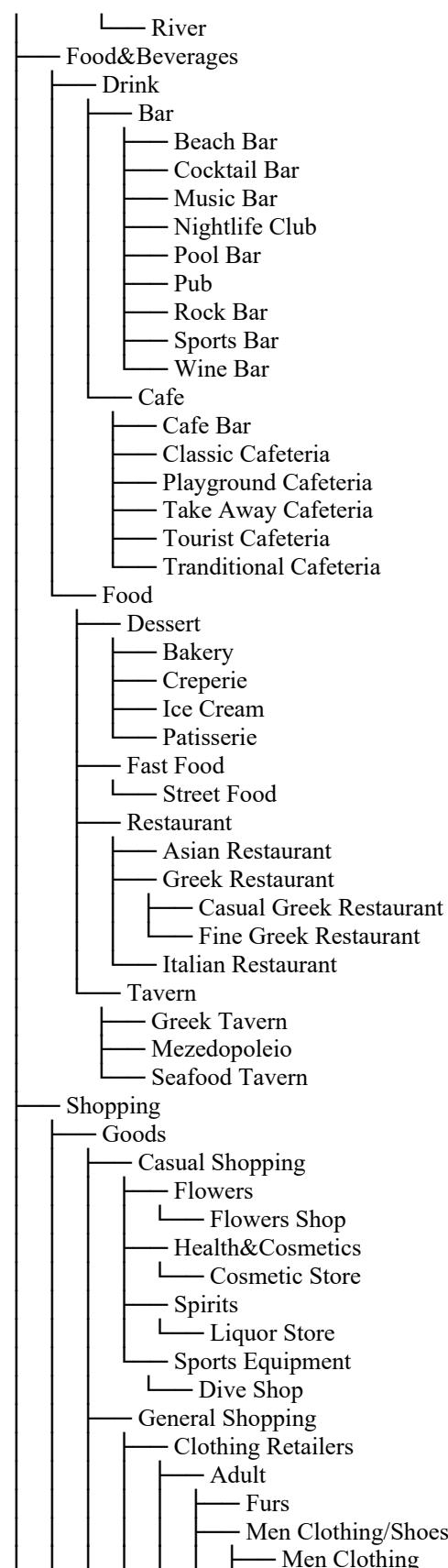
Informed Consent Statement: Not applicable.

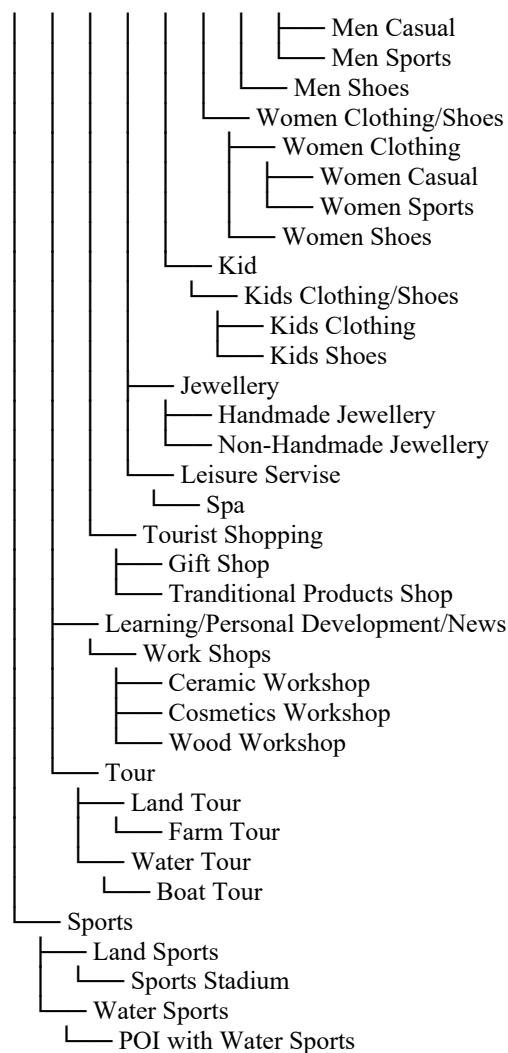
Data Availability Statement: The associated code and datasets are developed for the project *Visit Planner (ViP)* financed by the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH -CREATE-INNOVATE B cycle (project code: T2EDK-03135) and cannot be shared publicly at this point. The sharing of parts of the code or datasets could be approved, at the authors’ discretion, upon request.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Hierarchy Tree







Appendix B. Tourists Preferences Questionnaire



Agios Nikolaos Tourist Holiday Activities Survey

We at the Municipality of Agios Nikolaos strive to upgrade our hospitality services. Help us succeed in our mission by taking this survey. Thank you in advance.

1. Please state your nationality:

2. What is your gender? Male Female Prefer not to say

3. Would you like to tell us what age group you belong to?
 18-25 26-35 36-45 46-55 56-67 67+

4. Is Agios Nikolaos...
 ...your final holiday destination, as part of an organized tour?
 ...a port that is part of a cruise?
 ...a place you chose for a daily excursion while vacationing elsewhere?
 Other (Please specify):

5. Are you travelling:
 Alone As a couple With friends As a family (at least one child)
 With a group

6. Please rate (1-5, with 5=most important) the following criteria describing your ideal vacation:

Culture	Luxury accommodation and leisure
Sun & Sea	Nightlife
History / Archaeology	Gastronomy / Cuisine
Adventure / Sports	Sea Sports
Affordable prices	General shopping
Family-friendly activities & facilities	Shopping for local products
Rural tourism	Hiking / Trekking

Other (Please specify):



7. Duration of stay in Agios Nikolaos:

1 day 2 days 3-5 days 6-9 days 9 days or more

8. Activities that you chose/will choose during your stay in Agios Nikolaos:

Visits to archaeological sites, museums, churches	<input type="checkbox"/>	Visits to beaches	<input type="checkbox"/>
Visits to Restaurants, Bars, Clubs	<input type="checkbox"/>	Visit to Spinalonga	<input type="checkbox"/>
Daily tours	<input type="checkbox"/>	Shopping (souvenirs, clothes, traditional local products, etc.)	<input type="checkbox"/>
Extreme Sports	<input type="checkbox"/>	Relaxing by the lake	<input type="checkbox"/>

Other (Please specify):

9. Approximately how much money did you/will you spend on purchases of products and services during your stay in Agios Nikolaos (per day per person)?

Up to €50 €51-100 €101-150 €151-200 €200 or more

10. How much money did you/will you spend per category of expenses (per day per person) ?

(A) Up to €10, (B) €11-€20, (C) €21-€50, (D) €51-€100, (E) €101+

Food & Beverages	<input type="checkbox"/>	General Shopping	<input type="checkbox"/>
Nightlife	<input type="checkbox"/>	Local Products	<input type="checkbox"/>
Tours & Driving	<input type="checkbox"/>	Culture and Archaeological Sites	<input type="checkbox"/>
Accommodation	<input type="checkbox"/>		

References

- Veloso, C.M.; Walter, C.E.; Sousa, B.; Au-Yong-Oliveira, M.; Santos, V.; Valeri, M. Academic Tourism and Transport Services: Student Perceptions from a Social Responsibility Perspective. *Sustainability* **2021**, *13*, 8794. [[CrossRef](#)]
- Antunes, M.; Dias, A.; Gonçalves, F.; Sousa, B.; Pereira, L. Measuring Sustainable Tourism Lifestyle Entrepreneurship Orientation to Improve Tourist Experience. *Sustainability* **2023**, *15*, 1201. [[CrossRef](#)]
- Meleddu, M. Tourism, residents' welfare and economic choice: A literature review. *J. Econ. Surv.* **2014**, *28*, 376–399. [[CrossRef](#)]
- Ricci, F.; Rokach, L.; Shapira, B. Introduction to recommender systems handbook. In *Recommender Systems Handbook*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 1–35.
- Roy, D.; Dutta, M. A systematic review and research perspective on recommender systems. *J. Big Data* **2022**, *9*, 59. [[CrossRef](#)]
- Papadakis, H.; Papagrigoriou, A.; Panagiotakis, C.; Kosmas, E.; Fragopoulou, P. Collaborative filtering recommender systems taxonomy. *Knowl. Inf. Syst.* **2022**, *64*, 35–74. [[CrossRef](#)]
- Barbieri, N.; Costa, G.; Manco, G.; Ortale, R. Modeling item selection and relevance for accurate recommendations: A bayesian approach. In Proceedings of the 2011 ACM Conference on Recommender Systems, RecSys 2011, Chicago, IL, USA, 23–27 October 2011; Mobasher, B., Burke, R.D., Jannach, D., Adomavicius, G., Eds.; ACM: New York, NY, USA, 2011; pp. 21–28. [[CrossRef](#)]
- 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, Hong Kong, China, 12–16 October 2013; Association for Computing Machinery: New York, NY, USA, 2013; RecSys '13, pp. 221–228. [[CrossRef](#)]
- Tripolitakis, E.; Chalkiadakis, G. Probabilistic Topic Modeling, Reinforcement Learning, and Crowdsourcing for Personalized Recommendations. In *Proceedings of the Multi-Agent Systems and Agreement Technologies-14th European Conference, EUMAS 2016, and 4th International Conference, AT 2016, Valencia, Spain, 15–16 December 2016; Revised Selected Papers*; Pacheco, N.C., Carrascosa, C., Osman, N., Inglada, V.J., Eds.; Springer: Berlin/Heidelberg, Germany, 2016; Volume 10207, pp. 157–171. [[CrossRef](#)]
- Chaudhari, K.; Thakkar, A. A comprehensive survey on travel recommender systems. *Arch. Comput. Methods Eng.* **2020**, *27*, 1545–1571. [[CrossRef](#)]
- Streviniotis, E.; Chalkiadakis, G. Multiwinner election mechanisms for diverse personalized Bayesian recommendations for the tourism domain. In Proceedings of the 2022 Workshop on Recommenders in Tourism, RecTour, Seattle, WA, USA, 22 September 2022.
- Streviniotis, E.; Chalkiadakis, G. Preference Aggregation Mechanisms for a Tourism-Oriented Bayesian Recommender. In Proceedings of the PRIMA 2022: Principles and Practice of Multi-Agent Systems: 24th International Conference, Valencia, Spain, 16–18 November 2022; Proceedings; Springer: Berlin/Heidelberg, Germany 2022; pp. 331–346.
- Lord, P.W.; Stevens, R.D.; Brass, A.; Goble, C.A. Investigating semantic similarity measures across the Gene Ontology: The relationship between sequence and annotation. *Bioinformatics* **2003**, *19*, 1275–1283. [[CrossRef](#)]
- Pesquita, C.; Faria, D.; Falcao, A.O.; Lord, P.; Couto, F.M. Semantic similarity in biomedical ontologies. *PLoS Comput. Biol.* **2009**, *5*, 1–12. [[CrossRef](#)]
- Girardi, D.; Wartner, S.; Halmerbauer, G.; Ehrenmüller, M.; Kosorus, H.; Dreiseitl, S. Using concept hierarchies to improve calculation of patient similarity. *J. Biomed. Inform.* **2016**, *63*, 66–73. [[CrossRef](#)]
- Shenoy, M.K.; Shet, K.; Acharya, U.D. A new similarity measure for taxonomy based on edge counting. *Int. J. Web Semant. Technol.* **2012**, *3*, 23. [[CrossRef](#)]
- Wu, S.; Sun, F.; Zhang, W.; Xie, X.; Cui, B. Graph neural networks in recommender systems: A survey. *ACM Comput. Surv.* **2022**, *55*, 1–37. [[CrossRef](#)]
- Chen, Y.Y.; Cheng, A.J.; Hsu, W.H. Travel recommendation by mining people attributes and travel group types from community-contributed photos. *IEEE Trans. Multimed.* **2013**, *15*, 1283–1295. [[CrossRef](#)]
- Feng, J.; Xia, Z.; Feng, X.; Peng, J. RBPR: A hybrid model for the new user cold start problem in recommender systems. *Knowl.-Based Syst.* **2021**, *214*, 106732. [[CrossRef](#)]
- Zheng, X.; Luo, Y.; Xu, Z.; Yu, Q.; Lu, L. Tourism destination recommender system for the cold start problem. *KSII Trans. Internet Inf. Syst. (TIIS)* **2016**, *10*, 3192–3212.
- Panagiotakis, C.; Daskalaki, E.; Papadakis, H.; Fragopoulou, P. The tourist trip design problem with POI categories via an Expectation-Maximization based method. In Proceedings of the RecSys Workshop on Recommenders in Tourism (RecTour 2022), Seattle, WA, USA, 22 September 2022.
- Ziogas, I.P.; Streviniotis, E.; Papadakis, H.; Chalkiadakis, G. Content-based recommendations using similarity distance measures with application in the tourism domain. In Proceedings of the 12th SETN Conference on Artificial Intelligence, Corfu, Greece, 7–9 September 2022; pp. 1–10.
- Khatwani, S.; Chandak, M. Building Personalized and Non Personalized recommendation systems. In Proceedings of the 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), Pune, India, 9–10 September 2016; pp. 623–628. [[CrossRef](#)]
- Cremonesi, P.; Koren, Y.; Turrin, R. Performance of recommender algorithms on top-n recommendation tasks. In Proceedings of the fourth ACM conference on Recommender Systems, Barcelona, Spain, 26–30 September 2010; pp. 39–46.
- Umanets, A.; Ferreira, A.; Leite, N. GuideMe—A tourist guide with a recommender system and social interaction. *Procedia Technol.* **2014**, *17*, 407–414. [[CrossRef](#)]

26. Lim, K.H.; Chan, J.; Leckie, C.; Karunasekera, S. Personalized tour recommendation based on user interests and points of interest visit durations. In Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015.
27. Sertkan, M.; Neidhardt, J.; Werthner, H. PicTouRe-A Picture-Based Tourism Recommender. In Proceedings of the Fourteenth ACM Conference on Recommender Systems, Virtual Event, 22–26 September 2022; Association for Computing Machinery: New York, NY, USA, 2020; RecSys ’20, pp. 597–599. [[CrossRef](#)]
28. Sarkar, M.; Roy, A.; Agrebi, M.; AlQaheri, H. Exploring New Vista of Intelligent Recommendation Framework for Tourism Industries: An Itinerary through Big Data Paradigm. *Information* **2022**, *13*, 70. [[CrossRef](#)]
29. Riyahi, M.; Sohrabi, M.K. Providing effective recommendations in discussion groups using a new hybrid recommender system based on implicit ratings and semantic similarity. *Electron. Commer. Res. Appl.* **2020**, *40*, 100938. [[CrossRef](#)]
30. Fellbaum, C. WordNet. In *Theory and Applications of Ontology: Computer Applications*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 231–243.
31. Rendle, S.; Freudenthaler, C.; Gantner, Z.; Schmidt-Thieme, L. BPR: Bayesian personalized ranking from implicit feedback. *arXiv* **2012**, arXiv:1205.2618.
32. Sun, J.; Guo, W.; Zhang, D.; Zhang, Y.; Regol, F.; Hu, Y.; Guo, H.; Tang, R.; Yuan, H.; He, X.; et al. A framework for recommending accurate and diverse items using bayesian graph convolutional neural networks. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, 6–10 July 2020; pp. 2030–2039.
33. Benouaret, I.; Lenne, D. Personalizing the museum experience through context-aware recommendations. In Proceedings of the 2015 IEEE International Conference on Systems, Man, and Cybernetics, Hong Kong, China, 9–12 October 2015; pp. 743–748.
34. Chudoba, B. How Much Time are Respondents Willing to Spend on Your Survey? 2020. Available online: <https://tinyurl.com/6beyejwb> (accessed on 17 February 2023).
35. Wu, Z.; Palmer, M. Verb semantics and lexical selection. *arXiv* **1994**, arXiv:cmp-lg/9406033.
36. Lahitani, A.R.; Permanasari, A.E.; Setiawan, N.A. Cosine similarity to determine similarity measure: Study case in online essay assessment. In Proceedings of the 2016 4th International Conference on Cyber and IT Service Management, Bandung, Indonesia, 26–27 April 2016; pp. 1–6. [[CrossRef](#)]
37. Cox, D.R. The regression analysis of binary sequences. *J. R. Stat. Soc. Ser. B (Methodol.)* **1959**, *21*, 238. [[CrossRef](#)]
38. Celebi, M.E.; Kingravi, H.A.; Vela, P.A. A comparative study of efficient initialization methods for the k-means clustering algorithm. *Expert Syst. Appl.* **2013**, *40*, 200–210. [[CrossRef](#)]
39. Herlocker, J.L.; Konstan, J.A.; Terveen, L.G.; Riedl, J.T. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst. (TOIS)* **2004**, *22*, 5–53. [[CrossRef](#)]
40. Kurtenbach, G.; Sellen, A.J.; Buxton, W.A. Some articulatory and cognitive aspects of marking menus: An empirical study. *Hum. Comput. Interact.* **1993**, *8*, 1–23. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.