García García, I.; Sebastiá Tarín, L.; Onaindia De La Rivaherrera, E. (2011). On the design of individual and group recommender systems for tourism. Expert Systems with Applications. 38(6):7683-7692. doi:10.1016/j.eswa.2010.12.143.

On the design of Individual and Group Recommender Systems for Tourism

Inma Garcia [a,1,*], Laura Sebastia [a], Eva Onaindia [a]

[a] *Universidad Politecnica de Valencia, Spain*

Abstract

This paper presents a recommender system for tourism based on the tastes of the users, their demographic classification and the places they have visited in former trips. The system is able to offer recommendations for a single user or a group of users. The group recommendation is elicited out of the individual personal recommendations through the application of mechanisms such as aggregation and intersection. The elicitation mechanism is implemented as an extension of *e-Tourism*, a user-adapted tourism and leisure application whose main component is the *Generalist Recommender System Kernel (GRSK)*, a domain-independent taxonomy-driven recommender system.

Key words: Recommender Systems, Group Recommenders, Tourism

1 Introduction

Tourism is an activity strongly connected to the personal preferences and interests of people. Nowadays, more and more people realize the advantages of the new technologies for planning an agenda of leisure activities in a city [22] as an increasing number of companies and institutions offer tourist information easily accessible through web services. However, most of the existing tourism web sites can be regarded as booking services providers, and there is usually no recommendation on the available services - except for the typical user's

* Corresponding author. Tel: +34 963 870 000
Email addresses: `ingarcia@dsic.upv.es` (Inma Garcia), `lstarin@dsic.upv.es` (Laura Sebastia), `onaindia@dsic.upv.es` (Eva Onaindia).
[1] Partial support provided by Consolider Ingenio 2010 CSD2007-00022, Spanish Government Project MICINN TIN2008-6701-C03-01 and Valencian Government Project Prometeo 2008/051.

ratings that estimate the satisfaction level for the product and can serve as an indication for further customers. The reason why the successful application of movie or book recommendation techniques has not had such an impact in tourism is because modeling accurate tourism user profiles is a much harder task than in other application domains. Tourism is a much less frequent activity than movie-watching or book-purchasing and thus the number of available rated tourism items is also much smaller. On the other hand, the structure of a tourist product is itself far more complex than a book or a movie. Despite these difficulties, the current trend in travel, leisure and tourism web sites is to incorporate Recommender Systems (RS) to mimic the interaction with a human travel agent [4], and put the emphasis on the design of adaptive dialogs [16] aimed at eliciting the user preferences and requirements in order to come up with accurate tourism user profiles.

Among the various travel web services that use RS techniques, we can distinguish two types: (a) systems focused on recommending a tourist destination to the trip like *DieToRecs* [6], *ITR* [17] or *Trip@dvice* [18,21], and (b) systems focused on recommending a list of activities that a tourist can perform in a particular destination. The main difference between these two types of systems is that, whereas in the first case only a final result is requested [2], the second type of RS provide a list of activities and, more preferable, the construction of a tour or travel plan with such activities. For instance, *WebGuide* [7] generates personalized tour recommendations for the city of Heidelberg based on geographical information, information about points of interest and the individual users preferences and interests.

The system we describe in this paper, *e-Tourism*, falls within the second type of tourism web sites. *e-Tourism* is a tourist web-based RS that assists a user on the generation of a personalized tourist plan or agenda for the city of Valencia in Spain. *e-Tourism* eases the task of processing a large amount of information and selecting the most preferable activities for a particular user. Additionally, it also allows for the arrangement and schedule of the tourist activities by handling different sources of information like opening hours of places, distances between the places to visit or the time spent on the visit. This way we create a personalized agenda for the tourist with the recommended activities. This paper is particularly focused on the recommendation of the activities that are most likely of interest to the user, and we refer readers interested in the process of building the agenda to [19].

e-Tourism is also a group recommender system for tourism. Since traveling is an activity that usually involves a group of users (family, friends, etc.), travel

[2] Although *ITR* and *Trip@dvice* are specifically aimed at recommending a tourist destination, they do make use of a list of activities of interest to the user to obtain such a destination.

recommendations should meet the preferences of the majority of the group members [2]. One way to achieve this is that users discuss among themselves and arrive at a satisfactory agreement that combines the tastes and preferences of all the group members into a single set of preferences, thus allowing the system to elicit a recommendation as if they were a single user. However, this is a tedious and complicated task that requires the group members to previously agree on the way their particular preferences will be gathered together and combined into a single group profile. In order to alleviate the task of eliciting a group profile, group RS offer some sort of mechanisms for aggregating the individual models as to arrive at satisfactory recommendations for the whole group [10]; that is, by taking into account the interests and tastes of the group as a whole and by identifying the individual preferences, RS are capable of finding a compromise that is accepted by all the group members. This is the crucial point in a group RS because how individual preference specification and elicitation is managed to come up with the group model will determine the success of the recommendation [15,10].

To the best of our knowledge, there are not many group travel recommender systems. *CATS* [13] and *Travel Decision Forum* [11,10,12] are, for instance, two group RS aimed at recommending, specifically, a vacation destination, whereas *Intrigue* [2] assists a group of users in the organization of a tour and provides an interactive agenda for scheduling the tour. Both *CATS* and *Travel Decision Forum* build an individual user profile for each group member, and then maintain a group profile by means of a conversational mechanism. Specifically, in *CATS*, by critiquing a recommendation, the user can express a preference over a specific feature in line with their own personal requirements. The group profile is maintained by combining the individual user models and associating critiques with the users who contributed them. *Travel Decision Forum* uses animated characters to help the members of a group to agree on the organization of a vacation. Among the animated characters, there is a mediator who directs the interaction between the users. The objective of *Travel Decision Forum* is that users reach an agreement on the set of preferences (group profile) that the recommendation must fulfil. The degree of interest of a specific preference in the group profile is calculated out of the degree of interest of each member by using measures like the average, median, etc. or through an automatically designed non-manipulable aggregation mechanism. Once the group profile is created, the mediator asks each member of the group in turn whether the model can be accepted or not. By using the users critiques, the mediator reconfigures the preferences ratios, and the recommendation is done using the group preference model.

In *Intrigue*, individual participants are not described one by one but the system models the group as a set partitioned into a number of homogeneous subgroups, and their possibly conflicting individual preferences are separately represented. *Intrigue* elicits a set of preferences to define the subgroup requirements on the properties of tourist attractions, paying attention to those preferences possibly conflicting between subgroups. The group profile stores a relevance value to estimate the weight that the preferences of a member should have on the recommendation.

As a summary, we can conclude the three aforementioned group RS use aggregation methods to elicit the group profile, associating a weight or degree of interest to each preference in the group profile. These values are obtained according to the user critiques to the group preference model, the conflicts that may appear between them and, in general, through the interaction of the group members within the group travel recommender system.

Similarly to *CATS* or *Travel Decision Forum*, *e-Tourism* elicits an individual profile for each member in the group, and the group profile is maintained by combining the individual user models, thus giving rise to a set of preferences labeled with a degree of interest. Whereas the systems described above only use aggregation methods, our system also employs intersection mechanisms, a new functionality to elicit group recommendations such that no member in the group is specially promoted or harmed with the decisions; in other words, intersection mechanisms accounts for balanced decisions such that the preferences of all the group members are taken in account equally. Another distinguishing characteristic of our system is that, instead of making recommendations that directly match the group preferences, *e-Tourism* applies a hybrid recommendation technique by combining demographic, content-based recommendation and likes-based filtering. This way *e-Tourism* is always able to offer a recommendation, even when the user profile contains very little information. In comparison to other tourist group RS, *e-Tourism* provides a fully-automated mechanism for eliciting a group profile which does not require any interaction among the users, neither personally nor virtually. The resulting preferences of the *e-Tourism* preference elicitation could also be used as the starting point of a conversational mechanism in which users can further express and refine the group preference model.

This paper is organized as follows. The next section sketches the *e-Tourism* architecture. The main module of the architecture, the *Generalist Recommender System Kernel (GRSK)*, is fully detailed in section 3. As we will explain below, the recommendation process is divided into two steps, detailed in section 4 and section 5, respectively. The experimental results obtained from the evaluation of the GRSK, for both individual users and groups, are shown in section 6. We finally conclude and present some future work.

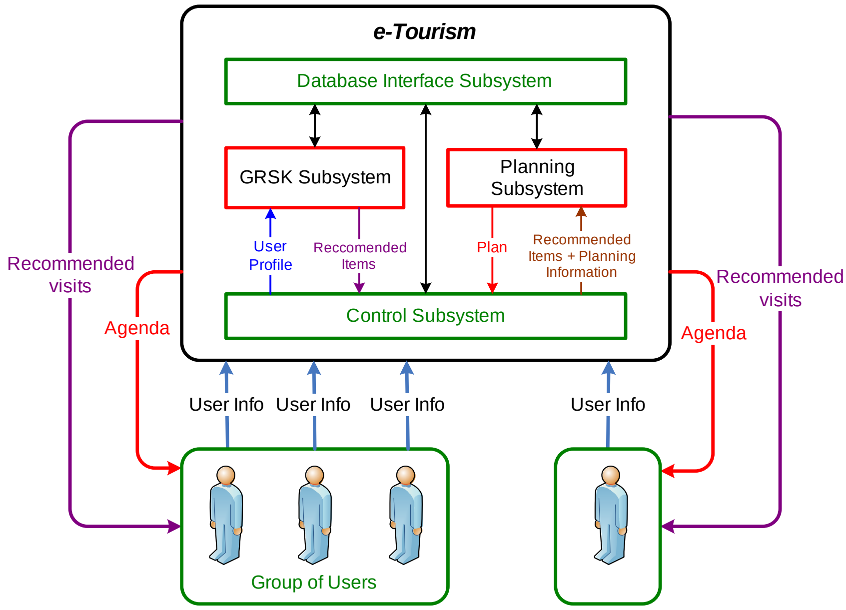

Fig. 1. *e-Tourism* system.

2 e-Tourism architecture

e-Tourism is a system aimed at recommending a list of tourist activities for a single tourist or group of tourists in a city, particularly, in the city of Valencia (Spain). It also provides a tour scheduling with the list of recommended activities complying with constraints such as the distances between places or the opening hours of places. *e-Tourism* is intended to be a service for foreigners and locals to become deeply familiar with the city and to plan leisure activities.

The *e-Tourism* architecture is shown in Figure 1. It is composed of four subsystems: the *Control* subsystem, the *Generalist Recommender System Kernel (GRSK)* subsystem, the *Planning* subsystem and the *Database Interface* subsystem. The Control subsystem acts as an user interface, initiates the execution of the other subsystems and centralizes the exchange of information. This includes converting the users request into a suitable recommendation query and show the list of recommended activities and the tour scheduling, i.e. the tourist plan. The Database Interface processes the queries coming from the rest of modules in the system. The two most important modules in *e-Tourism* are the GRSK and the Planning subsystems. The GRSK is a general-purpose recommender system in charge of generating the list of recommended activities, and the Planning subsystem schedules the selected activities thus building the tourist plan or agenda [19].

e-Tourism works as follows. The first step is to **build the individual user profile**. First, the user registers in the system and enters his personal details

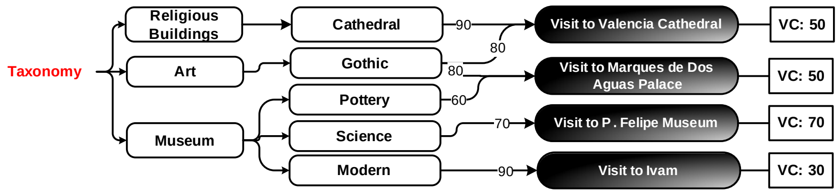

Fig. 2. GRSK taxonomy.

and general likes. In case of a group, it is necessary to give the users that form the group, who must be already registered in the system. The second step, executed by the GRSK subsystem, is to **generate a list of activities** that are likely of interest to the user or the group of users. The last step is to **schedule the selected activities** and return the tourist plan or agenda, as indicated in Figure 1. As we commented before, this paper does not deal with the problem of building the agenda and interested readers are referred to [19].

3 The Generalist Recommender System Kernel (GRSK)

The *Generalist Recommender System Kernel (GRSK)* is the core of *e-Tourism*. The two main tasks of the GRSK are the analysis of the users profile to come up with a preference model, and the generation of the list of activities to be recommended to a single user or a group of users. These two tasks are described in detail in sections 4 and 5, respectively.

In this section, we present the different elements and components of the GRSK, namely the hierarchical structure of the data handled in the GRSK, the input information that users provide when building their profiles and the components that make up the GRSK architecture.

3.1 The GRSK data

3.1.1 Taxonomy

The GRSK relies on the use of a taxonomy to represent the user's likes and the items to recommend (see Figure 2). It has been designed to be *generalist*, i.e. independent of the current catalog of items to recommend. Therefore, the GRSK can work with any application domain provided that the data of the new domain are defined through a taxonomy representation.

The entities in a **taxonomy** are arranged in a hierarchical structure connected through an *is-a* relationship in which the classification levels become more specific towards the bottom. In a GRSK taxonomy, entities represent the

features (F) that are commonly managed in a tourism domain like *Gothic Art*, *Museums*, *Religious Buildings*, etc. as Figure 2 shows. The leaf nodes of the taxonomy represent the **items** to recommend; they are categorized by the lowest-level or most specific feature in the hierarchy. The edges linking an item to a feature are associated a value to indicate the **degree of interest** of the item (activity in the tourism taxonomy) under the corresponding feature, i.e. as a member of the category denoted by the feature. An item can also be categorized by more than one feature in the taxonomy. For instance, in Figure 2, the item *Visit to Marques de Dos Aguas Palace* is categorized with 80% of interest as *Gothic Art* and with 60% of interest as a *Pottery Museum*.

Items are described by means of a list of tuples which represent all the incoming edges of a leaf node. A tuple is of the form (i, f, d^{if}), where i is the item, $f \in F$ is a feature defined in the taxonomy such that there is an edge connecting f and i, and $d^{if} \in [0, 100]$ is the degree of interest of the item i under the feature f. Additionally, items are associated a numeric value VC^i (**visit counter**) to represent how popular the item i is among users; this value indicates how many times the item i has been visited by the users when it was recommended.

3.1.2 Input information of the user profile

The GRSK records a profile for each user that contains the user tastes and general likes as well as his historical interaction with the system. The input information on the personal data and general likes of the user is entered only on the first visit, i.e. when the user registers in the system for the first time. The information obtained during the interaction of the user with the system after the visit will be further used to better capture his/her likes and update the profile.

The **profile** of a given user u stores the following information:

1) Personal and demographic details like the age, the gender, the family or the country.
2) The general-likes model of the user (GL^u) is a list of the features in the taxonomy which the user u is interested in along with the user ratings for those features. GL^u is represented by a list of tuples of the form $GL^u = \{(u, f, r^{uf})\}$, where $f \in F$, and $r^{uf} \in [0, 100]$ is the rating given by the user u to the feature f.
3) Information about the historical interaction of the user with the recommender system, namely the set of items the user has been recommended and his degree of satisfaction with the recommended items. A rated item is described by the tuple (u, i, r^{ui}), where u and i denote the user and the recommended item, respectively, and $r^{ui} \in [0, 100]$ is the rating given by the user u to the item i (degree of satisfaction of u with the item i).

In case of a **group of users**, all the individuals must be previously registered in the system. The GRSK will consider the profile of each user in the group to maintain a group profile and give the group recommendation. For subsequent visits of the group, the GRSK will make up again the group profile so as to take into account the ratings of the users to past visits and thus work with a more accurate group profile.

3.1.3 Information exchange during the recommendation process

Each time a user or a group enters *e-Tourism* for a new visit, the system creates a **recommendation query**, a data structure which represents the requested recommendation. A recommendation query contains the users profiles and the maximum number of items the users want to be recommended.

The result of the recommendation process is a **list of recommended items**, or more specifically a list of activities to perform in the city, which we will call RI. RI is a set of tuples of the form $RI = \{(u/G, i, d^{u/G,i})\}$, where u/G represents the user or the group of users, i is a recommended item, and $d^{u/G,i}$ is the estimated degree of interest of the user u or group G in the item i. After the visit, users will individually rate the proposed activities or items. *e-Tourism* will use this feedback to increase the accuracy of the users profile and thus make better estimations of the items that are likely to be of interest to the user in future visits.

3.2 The GRSK architecture

Figure 3 shows an sketch of the GRSK architecture, which is composed of the following modules:

- The **Engine** module is the core of the GRSK. It is in charge of managing the recommendation query and of generating and updating the users profiles.
- The **Single User/Group Manager** controls the recommendation process. This module receives the profiles created by the users and sends them one by one to the module that applies the Basic Recommendation Techniques (see next paragraph). As a result from this application, the manager receives a set of preferences for each user. In our context, a preference is a feature in the taxonomy of likely interest to the user which is calculated by any of the basic recommendation techniques. If the user is a group, the manager invokes the Group Preferences Manager to elicit the group preferences out of the individual preferences. Finally, it will call the Items Selector and the Hybrid RS in order to retrieve the list of items that best match the user/group preferences.

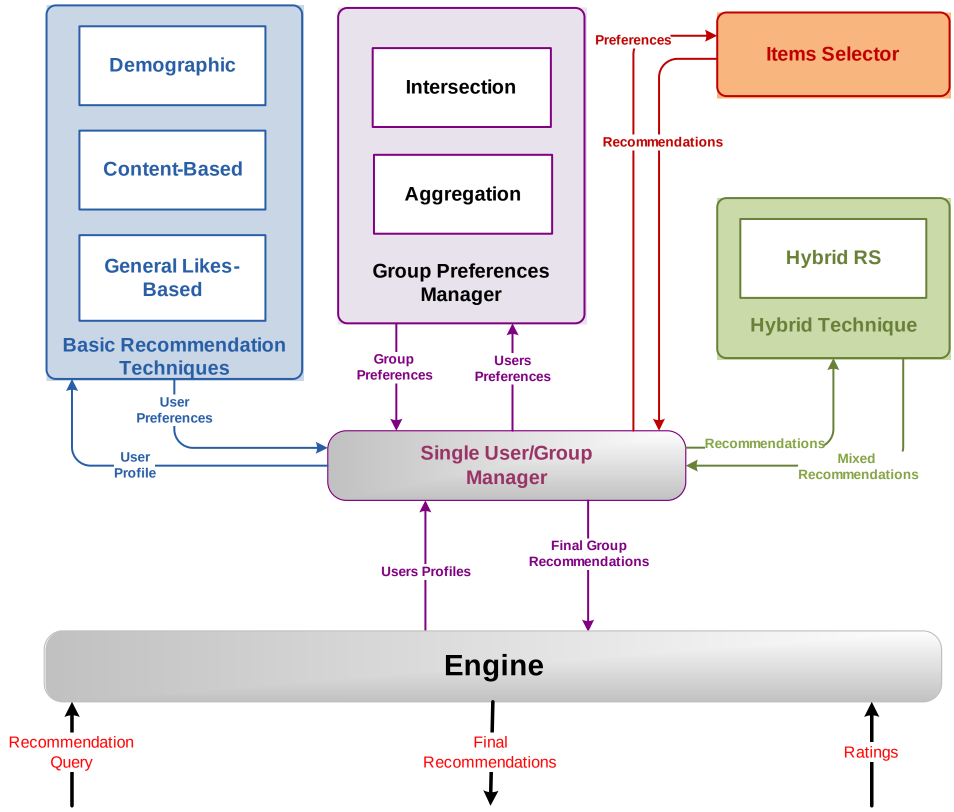

Fig. 3. GRSK architecture.

- The **Basic Recommendation Techniques** (BRTs) module applies recommending techniques like demographic RS [3], content-based RS [3] and general likes-based filtering [9] to elicit the preferences that embody the individual likes of each user. For a given user, each of the three techniques technique creates a different list of preferences according to the parameters and data handled by the technique. Therefore, the three lists of user preferences are independent of each other.
- The **Group Preferences Manager** (GPM). This module is only invoked when the user is a group, and it offers two mechanisms such as aggregation and intersection to elicit the group preferences out of the individual preferences. Actually, the GPM generates three lists - the group demographic preferences, the group content-based preferences and the group filtering preferences.
- The **Items Selector** receives the three lists of preferences and, for each list, it returns the set of items (activities) that better match the elements in the list.
- The **Hybrid Technique** module gathers together the three lists of items returned by the Items Selector and creates a single list that embodies the final user/group recommended items. Since each BRT exhibits some advantages and disadvantages [1], a common solution adopted by many RS is to combine various techniques into a hybrid RS [14,3], thus alleviating the

limitations of one technique with the advantages of the others. The GRSK applies a mixed hybrid recommendation technique that will be detailed in section 5.2.

4 Analysis of the user profile

The first step in the recommendation process is to analyze the user profile and elicit the list of preferences for each user. When working with a group of users, this elicitation mechanism is performed as many times as users in the group.

Definition 1 *A* preference *is a tuple of the form* (u, f, d^{uf}) *where* u *denotes the user,* f *a feature in the taxonomy and* $d^{uf} \in [0, 100]$ *is the interest degree of the user* u *in the feature* f*.*

A preference is a feature in the taxonomy with a interest-degree of d^{uf} for a user u selected by one of the three basic recommendation techniques. The value d^{uf} may be the rating value directly provided by the user u to the feature f at the time of creating the general-likes model of his/her profile (tuples of the form (u, f, r^{uf}), and thereby $d^{uf} = r^{uf}$), or may be a value computed out of all of the items described under the feature f, specifically from the interest-degree of the items under the feature f and the ratings of the user to such items, i.e. computed from tuples (i, f, d^{if}) and (u, i, r^{ui}).

It is important to note that, unlike most tourist RS, *e-Tourism* does not initially work with the items or activities that will be later recommended to the user. In contrast, rather than using items, *e-Tourism* makes use of the concept of feature to elicit the user preference model, which is a more general and flexible entity. This makes the GRSK able to work with any application domain as long as the data can be represented through a taxonomy.

As Figure 4 shows, the first part of this elicitation mechanism is to invoke the **Basic Recommendation Techniques (BRTs)** module. The application of the three recommendation techniques return each a different list of individual preferences for a user. Therefore, after this stage, we will have for each user three lists of individual preferences, namely a demographic, content-based and general-likes-based list [3] , which describe the usual tastes of the user. The three lists are then passed to the **Items Selector**, which will select the items that best match these preferences (see Figure 3).

[3] We opted for these techniques because we considered them more suitable for our current domain (tourism and leisure). We do not use the collaborative recommendation, which is the most widely used recommendation technique, because it presents some difficulties to be applied in this domain [5].

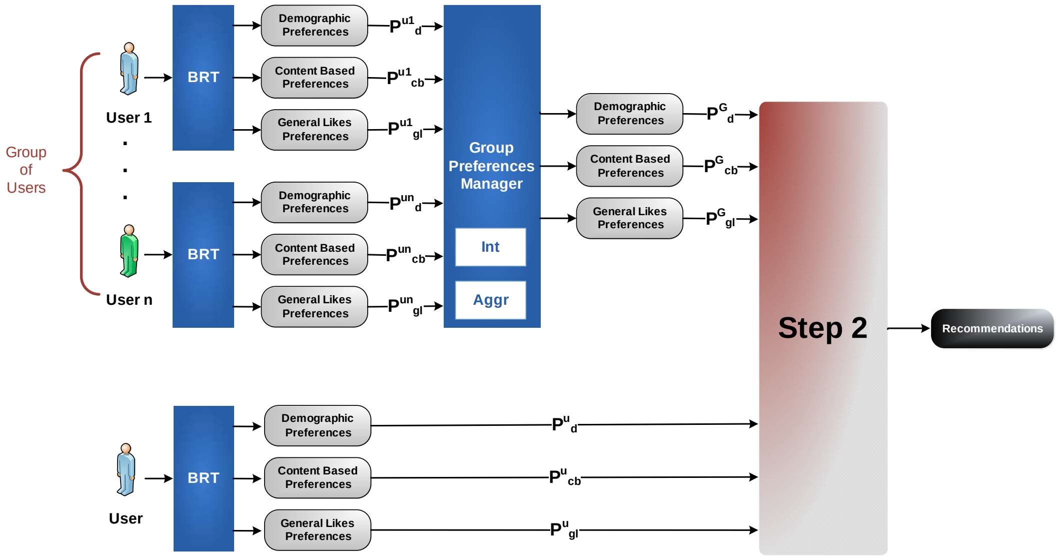

Fig. 4. Sketch of the analysis of the user profile.

In case of a group, prior to call the Items Selector, the system will invoke the **Group Preferences Manager** which is the module in charge of eliciting the preferences of the whole group. Thus, the behavior of the Items Selector and the Hybrid RS is independent from the fact that we are dealing with a single user or a group of users.

4.1 Basic Recommendation Techniques

As we said above, each recommendation technique generates an independent list of preferences for each user and hence the lists may contain different features or the same feature with different degrees of interest. We will call these lists P^u_d for the demographic preference list, P^u_{cb} for the content-based preference list, and P^u_{gl} for the general-likes-based preference list.

The **demographic BRT** classifies the user u into a demographic category according to his profile details. For example, a person with children is classified into a different category than a retiree as they will likely have different preferences. We opted for a demographic BRT because it is a good alternative to solve the problem of the *new user* as it is always able to give a recommendation. In addition, it can recommend items which contain characteristics different from other previously recommended items.

The **content-based BRT** computes a set of preferences by taking into account the items that have been previously rated by the user (historical interaction). Let f be a feature and I a list of items described under the feature f in the taxonomy; I will be a list of tuples of the form (i, f, d^{if}) for a particular feature f. Let $RT^u = \{(u, i, r^{ui})\}$ be the set of items valued by a user u with respective ratings of r^{ui}; a preference (u, f, d^{uf}) is added to the list P^u_{cb} where:

$$d^{uf} = \frac{\sum_{\forall i \in I \cap RT^u} d^{if} * r^{ui}}{|RT^u|}$$

The value d^{uf} denotes the interest-degree of a user u for the items described under the feature f amongst the whole set of items rated by u. The use of a content-based technique allows us to recommend items similar to the ones already accepted by the user thus increasing the user satisfaction. For example, if the user likes visiting museums, the system will tend recommending visits to other museums.

The **general-likes-based BRT** is an information filtering technique that works with the general-likes model specified by the user in his profile (GL^u). In this case, the set of preferences P^u_{gl} is simply built as $P^u_{gl} = GL^u$; that is, the interest-degree of the preferences in P^u_{gl} will be the ratings given by the user to that particular feature in his profile $(d^{uf} = r^{uf})$.

4.2 Group Preferences Manager

The elicitation of the group preferences is managed by the **Group Preferences Manager (GPM)** (Figure 4). The GPM is fed with the three lists of individual preferences of each user and returns three lists of group preferences. The individual preferences of a user u are denoted by (P^u_d, P^u_{cb}, P^u_{gl}), where a preference has the usual form of (u, f, d^{uf}). From these lists, the GPM returns P^G_d, the demographic group preference list, P^G_{cb}, the content-based group preference list, and P^G_{gl}, the general-likes group preference list. The GPM makes use of two disjunctive methods to elicit the group preferences: **aggregation** and **intersection**. These methods, detailed in the following sections, differ on the way the lists of individual preferences are combined.

4.2.1 Aggregation GRT

The aggregation mechanism is a common technique that has been used in various group RS (see section 1). Aggregation gathers the preferences, computed by the BRT modules, of all members in the group to make up a single set of preferences. More specifically, the individual preferences returned by each BRT are used to create a single set of preferences for each type of recommendation (P^G_d, P^G_{cb}, P^G_{gl}). We denote by P^G_{brt} the set of preferences corresponding to a particular BRT, where $brt \in \{d, cb, gl\}$.

P^G_{brt} is the result of aggregating the preferences returned by the corresponding

Preferences	D	CB	GL
User 1	(SM,70)	(PM, 40)	
User 2	(G,80), (SM,50)		(C, 70)
User 3	(SM,30)	(MM,65)	
Intersection	(SM,50)		
Aggregation	(G,80), (SM,50)	(PM, 40), (MM,65)	(C, 70)

C: Cathedral
G: Gothic
PM: Pottery Museum
MM: Modern Museum
SM: Science Museum

Fig. 5. Example of elicitation of the preferences of a group.

BRT for at least one user in the group, that is, a feature f belongs to P^G_{brt} if it belongs at least to one of the P^u_{brt} lists. The interest-degree of a group preference d^{Gf} is calculated as the ***average value*** [4] of the interest-degree of the users in G for the feature f. More formally:

$$P^G_{brt} = \{(G, f, d^{Gf}) : \exists (u, f, d^{uf}) \in \bigcup_{\forall u \in G} P^u_{brt}\}, \text{where } d^{Gf} = avg(d^{uf})$$

As the results presented in section 6 will show, aggregating preferences does not necessarily account for the preferences of the group as a whole.

4.2.2 Intersection GRT

The intersection mechanism is introduced as a counterpoint of the aggregation mechanism. This method finds the preferences that are shared by all the members in the group and make up the group preferences. More formally:

$$P^G_{brt} = \{(G, f, d^{Gf}) : \exists (u, f, d^{uf}) \in \bigcap_{\forall u \in G} P^u_{brt}\}, \text{where } d^{Gf} = avg(d^{uf})$$

As the results presented in section 6 will show, the advantage of this mechanism is that all of the users in the group will be equally satisfied with the resulting group profile. However, the risk of using intersection is that we might end up with an empty list of preferences if the group is rather heterogeneous.

4.2.3 Example of elicitation of group preferences

Figure 5 shows an example of the elicitation of the preferences of a group by using aggregation and intersection. This example is based on the taxonomy in Figure 2. The first three rows in the table show, for each user, the lists of preferences computed by each BRT. The intersection method obtains only one preference (*Science Museum -SM-*) because it is the only feature shared by all

[4] For the sake of simplicity, we compute the average, but it could also be defined as the maximum, the median or the addition of the same set of values.

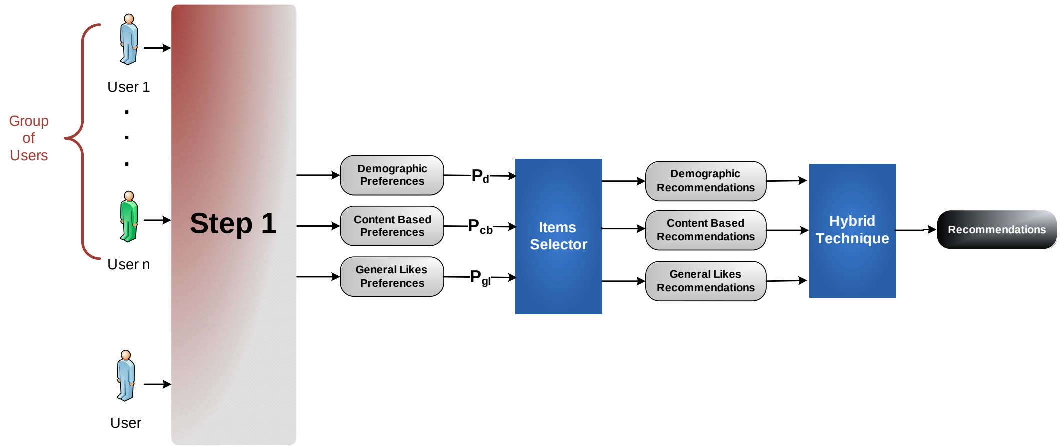

Fig. 6. Sketch of the computation of the recommended items.

the group members. On the other hand, the aggregation mechanism creates one list per BRT with the individual preferences of all the users. For instance, the d^{Gf} value associated to the feature *Science Museum* is computed as the average of the d^{uf} values of all the group members for such a feature (user 1 $(SM, 70)$, user 2 $(SM, 50)$ and user 3 $(SM, 30)$).

5 Calculation of the recommended items

The second step in the recommendation process is to call the Items Selector to select, among all of the items in the taxonomy, those ones that best match the preferences of the lists P_d^G, P_{cb}^G and P_{gl}^G. Regardless the recommendation is for a single user or for a group [5], the outcome of the user profile analysis is a set of three lists of preferences, one per BRT. This information feeds the **Items Selector**, which retrieves the items that match the preferences in the lists P_d^G, P_{cb}^G and P_{gl}^G. Afterwards, the **Hybrid Technique** will apply a mixed hybrid recommendation [3] in order to obtain a single list of ranked recommendations. Figure 6 sketches the process for calculating the recommended items.

The result of this step is, therefore, a list of ranked items that we will denote as $RI^G = \{(G, i, d^{Gi})\}$, where G is the group, i is the item, and d^{Gi} is the interest-degree of the item i for the group G. Section 5.1 explains how to select the items i, and section 5.2 is devoted to the calculation of the value d^{Gi}.

[5] For the sake of simplicity, in the following sections, we assume that if we are dealing with a single user u, then we have a group with only one member: $G = \{u\}$.

5.1 Items Selector

The method for selecting an item is quite simple: an item i represented by the tuple (i, f, d^{if}) matches a preference in P^G_{brt} if there is a tuple (G, f, d^{Gf}) in P^G_{brt} such that the item has not previously rated by any user in the group, i.e. none of the group members have performed yet such an activity in the city. Formally, these two conditions can be expressed as follows:

(1) $\exists(i, f, d^{if}) \in taxonomy \wedge \exists(G, f, d^{Gf}) \in P^G_{brt}$
(2) $\nexists(u, i, r^{ui}) \in RT^u \;\; \forall u \in G$

The outcome of the Items Selector is a set of three lists of ranked items, one list per BRT.

5.2 Hybrid technique

The three lists of recommended items computed by the Items Selector are then processed by the Hybrid Technique, which applies a mixed hybrid recommendation [3] and returns a single list of ranked items (RI^G). By handling these lists of items independently, we give much more flexibility to the GRSK, because any other hybrid technique can be used by simply replacing one component by another. The value d^{Gi} of a tuple in RI^G is calculated as follows:

$$d^{Gi} = percentile(VC^i) + avg_{\forall f}(d^{if} + d^{Gf})$$

where $percentile(VC^i)$ refers to the percentile of the visit counter of i (VC^i) with respect to the whole set of items visited by the users. The second part of the formula considers the average interest-degree of all the features that describe the item i in both the taxonomy (d^{if}) and in the group preferences (d^{Gf}).

The Hybrid Technique computes all of the items that match the group preferences and retrieves the best ranked elements. Assuming the group has solicited N recommendations, the hybrid technique will select the N best ranked items and will insert a tuple of the form (G, i, d^{Gi}) in RI^G for each of these N best items.

5.3 Example of recommendation

We continue here with the example introduced in section 4.2.3. Figure 7 shows that, when using intersection, the system will only recommend items described

Preferences	D	CB	GL
Intersection	(SM,50)		
Aggregation	(G,80), (SM,50)	(PM, 40), (MM,65)	(C, 70)

Items recommended			
Items Intersection		Items Aggregation	
P. Felipe Museum	190	Valencia Cathedral	210
		P. Felipe Museum	190
		Ivam Museum	185

C: Cathedral
G: Gothic
PM: Pottery Museum
MM: Modern Museum
SM: Science Museum

Fig. 7. Example of the calculation of the items to recommend.

under the feature *Science Museum*; in the taxonomy of Figure 2, only one item is associated to this feature, *Visit to Prince Felipe Museum*. Assuming that $percentile(VC^{PrinceFelipeMuseum})$ is 70, the estimated degree of interest of this item is computed as ($d^{if} = 70$ and $d^{Gf} = 50$): $d^{G,PrinceFelipeMuseum} = 70 + avg(70 + 50) = 190$. On the other hand, when using aggregation, all of the items will be recommended, and the order of the final recommendations will depend on the degree of interest of each item. For example, in this case, the estimated interest-degree of *Visit to Valencia Cathedral* is computed as: $d^{G,ValenciaCathedral} = 50 + avg(80 + 80, 90 + 70) = 210$; this item is described under the features *Gothic Art* and *Cathedral* with d^{if} values of 80 and 90, respectively. Assuming the group has solicited three items, the three best ranked items are recommended (*Valencia Cathedral*, *Prince Felipe Museum* and *IVAM Museum*). Note that the *IVAM Museum* is recommended although only one of the users has *Modern Museum* among his/her preferences.

6 Experimental results

In this section, we detail the experiments we carried out to validate our approach both for a single user (section 6.1) and for a group of users (section 6.2).

Due to the fact that we are working with our own domain, our first task was to obtain data from real users. We prepared a questionnaire with questions about general preferences, demographic data, visited places and the degree of satisfaction of the realized visits. The questionnaire was filled in by 60 people, and this information was used to create 60 users in our database; 50 out of the 60 users were used to train the system and the remaining 10 users were used as test users.

6.1 Single user recommendation results

In order to test the GRSK for a single user, we selected two classical Information Retrieval metrics: *precision* and *recall*. In an Information Retrieval scenario, precision is defined as the number of retrieved relevant items divided by the total number of items retrieved by the search; and recall is defined as the number of retrieved relevant items divided by the total number of existing relevant items. That is, precision represents the probability that a retrieved item is relevant to the user, and recall is the probability that a relevant item is retrieved by the search.

Specifically, we call Ns the number of retrieved items by the GRSK, that is, the number of recommendations solicited by the user/group. The number of relevant items is denoted by Nr. We consider as relevant items those places that the test users have marked as visited with a positive degree of satisfaction in the questionnaire. Finally, Nrs is the number of relevant items retrieved in the recommendation, that is, $Nrs = Nr \cap Ns$. Then, precision and recall are calculated as follows:

$$P = \frac{Nrs}{Ns} \qquad\qquad R = \frac{Nrs}{Nr}$$

Often, there is an inverse relationship between P and R, where it is possible to increase one at the cost of reducing the other. For example, R can be increased by increasing Ns, at the cost of increasing the number of irrelevant items retrieved and thus decreasing P. For this reason, P and R ratios are not discussed in isolation. Instead, both are usually combined into a single measure, such as the $F - measure$:

$$F = \frac{2 * P * R}{P + R}$$

We ran our experiments in terms of two parameters, Ns, the number of retrieved items, and the information about past visits in the user profile. As for Ns, we ran tests with $Ns = 10$ and $Ns = 25$. The list of retrieved items was the same in both experiments, but, in the first case only the first 10 items were considered whereas in the second case we considered the first 25 items. Regarding the second parameter, we took into account four levels of historical information in the user profile; a new user ($H = 0$) and user profiles that store 25% ($H = 25$), 50% ($H = 50$) and 75% ($H = 75$) of (randomly selected) past visits, respectively. Figures 8 and 9 show the $F - value$ obtained for the 10 test users (X axis) when $Ns = 10$ and $Ns = 25$, respectively.

In Figure 8 ($Ns = 10$), when dealing with a new user ($H = 0$), we can conclude that the quality of the recommendations measured by the $F - value$

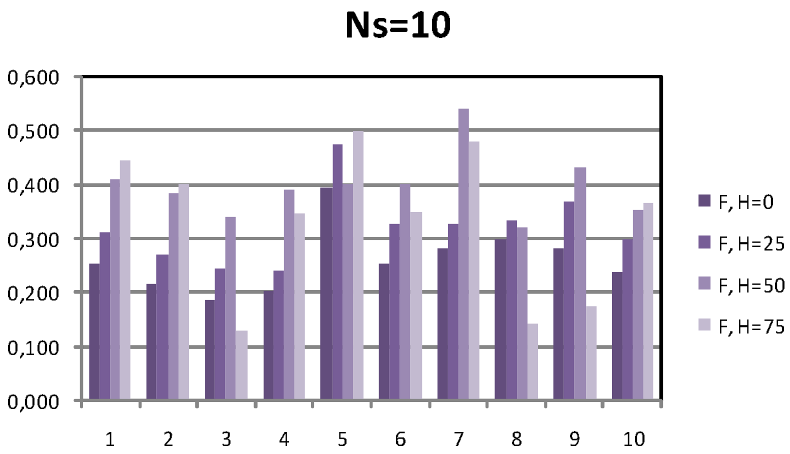

Fig. 8. Comparison of the F value obtained for 10 users when $Ns = 10$ and for the four degrees of historical information.

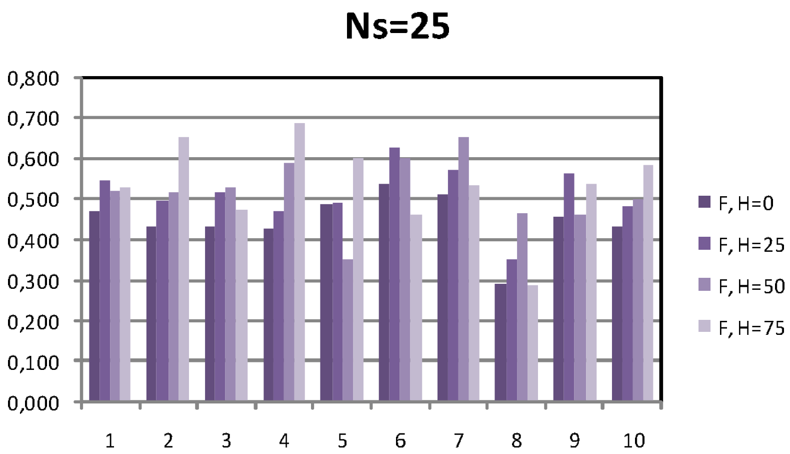

Fig. 9. Comparison of the F value obtained for 10 users when $Ns = 25$ and for the four degrees of historical information.

is neither good nor bad. As expected, when the information provided to the system increases ($H = 25$, $H = 50$), the GRSK improves the quality of the recommendations. However, in some of the cases in which the user feedback is rather high ($H = 75$), the quality of the recommendation given by the $F - value$ worsens. This is because the database does not contain a large number of items and, therefore, the GRSK is not able to recommend places other than those ones already visited by the user.

In Figure 9 ($Ns = 25$), the general impression is quite similar. However, in this case, the F-values are better because, although the precision is a bit lower, the recall increases in a higher order. Here again, the more feedback, the better the quality of the recommendation, and, unlike the previous case, the worsening

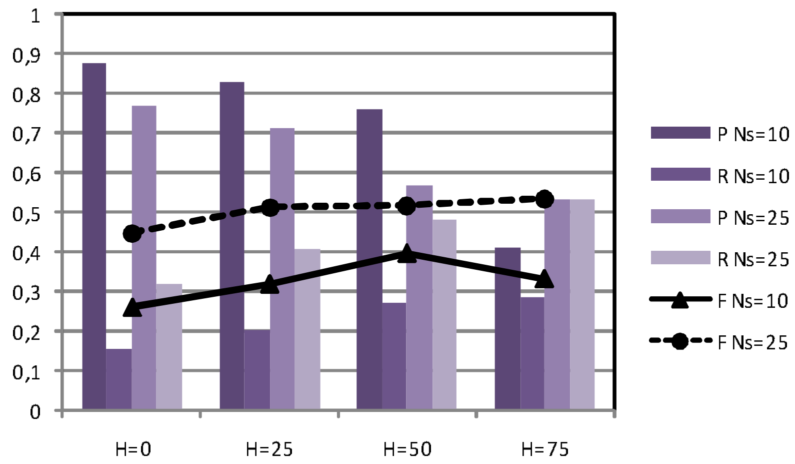

Fig. 10. Comparison of the P, R and F values obtained when $Ns = 10$ and $Ns = 25$ and for the four degrees of historical information.

in the case of $H = 75$ is not as noticeable.

Figure 10 shows a comparison between the average of precision (P) and recall (R) for the four the different cases of user feedback. When $Ns = 10$, the difference between the precision and the recall is remarkable, and the precision decreases as the recall increases, as expected. However, when $Ns = 25$, this difference is not so noticeable. In both cases, the more feedback, the higher the $F - value$ in average, except when $Ns = 10$ and $H = 75$, for the same reason as explained above.

In summary, the more feedback, i.e., the GRSK has more knowledge about the past visits of the users, the better the quality of the recommendation. However, when a certain level of feedback is reached, a very large number of rated items may worsen the recommendation quality due to the lack of new items to recommend.

6.2 Group recommendation results

Unlike individual recommendations, when dealing with groups, the most important issue is to obtain recommendations as satisfactory as possible for all the group members. Through the experimental setup presented in this section, we intend to analyze which of the two mechanisms between aggregation or intersection returns the best recommendations from the perspective of the satisfaction of the whole group.

Let RI^u be the recommendation for a single user u, such that each element in RI^u has the form (u, i, d^{ui}), where i is the recommended item ($i \in I$) and d^{ui}

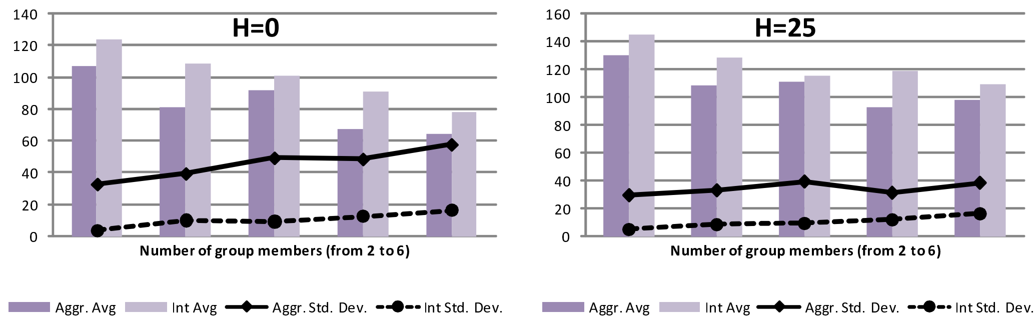

Fig. 11. Comparison of the quality of the recommendations (Aggr and Int) - $H = 0$ and $H = 25$.

is the estimated degree of interest of the item i for user u.

If the value d^{ui} is unknown for a certain item i, then it means the item has not been recommended to the user and thereby d^{ui} is set equal to 0. Given a recommendation RI^G for a group G such that $u \in G$,the **utility** of such a recommendation for a user u is calculated as follows:

$$U_u^G = \frac{\sum_{\forall i \in RI^G} d^{ui}}{Ns}$$

This value gives a measure of the **precision** of the group recommendation for each user in the group. In order to analyze the quality of the recommendations, we consider the average and the standard deviation (dispersion) on the utility over all the group members: $\mu_G(U_u^G)$ and $\sigma_G(U_u^G), \forall u \in G$.

We performed experiments with aggregation (Aggr) and intersection (Int) by using groups of different size ranging from 2 to 6 members; the number of requested recommendations or retrieved items (Ns) was set to 10 in all cases. We also ran the experiments with four levels of feedback or historical information, from 0% to 75%. Figure 11 shows the results obtained for $H = 0$ and $H = 25$, and Figure 12 shows the results obtained for $H = 50$ and $H = 75$. Bars represent the utility in average of the recommendations for each group size (2,3,4,5 and 6 members) and elicitation mechanism. Likewise, the points in the lines determine the dispersion level in average for each group size and elicitation mechanism.

In all the experiments, it can be observed that the utility in average is better with the intersection mechanism for all group sizes and levels of feedback. Also, the dispersion on the utility is lower when using intersection. The reason behind is that the intersection considers the preferences that satisfy a larger number of users, whereas the aggregation recommends the most prioritized items for *at least one* member in the group, which obviously does not imply to be for *all* the group members. Therefore, we can affirm that the utility or

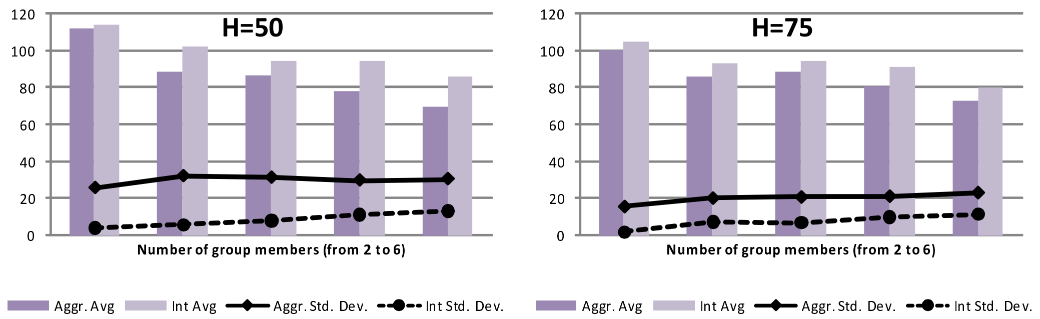

Fig. 12. Comparison of the quality of the recommendations (Aggr and Int) - $H = 50$ and $H = 75$.

degree of satisfaction with the recommendations obtained by the intersection mechanism is higher than with the solutions of the aggregation mechanism. In addition, the intersection recommendations have also a lower degree of dispersion, which is interpreted as all the group members are equally satisfied.

On the other hand, as the number of members in the group increases, the utility in average usually decreases for both intersection and aggregation. Clearly, it is more difficult to find satisfying group recommendations for large groups. As for the user feedback, we can conclude that the best results are obtained for $H = 25$; in the rest of cases, we obtain lower utility values that are quite similar among them. This occurs because, as the number of rated items increases, it becomes more difficult to find items that have not been previously visited by any user in the group.

7 Conclusions and further work

e-Tourism is a web-based service to make recommendations about personalized tourist tours in the city of Valencia (Spain) for a single user or a group of users. The component in charge of the recommendation process is the GRSK, a taxonomy-driven domain-independent recommendation kernel. Single user recommendations are computed according to the user preferences by using a hybrid recommendation technique that mixes three basic recommendation techniques: demographic, content-based and general likes filtering. Group preferences are elicited out of the individual preferences through the application of the intersection and aggregation mechanisms. Finally, the list of elicited preferences yield to a list of items to recommend. The evaluation of this process shows that the intersection mechanism obtains better results because it brings together the preferences of all the group members.

GRSK has been designed as a generalist recommender system kernel. This allows us to easily add new basic, hybrid or group recommendation techniques, thus providing the system a great flexibility.

Finally, we are also working in the use of agreement techniques to obtain group recommendations [8]. The members of the group are modeled as agents who attempt achieving a reconciled solution for the whole group maximizing the user satisfaction. The inclusion of this technique will allow us to account for more sophisticated user behaviors into the group.

References

[1] Adomavicius G., Tuzhilin A., Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions, IEEE Transactions on Knowledge and Data Engineering 17 (6) (2005) 734–749.

[2] Ardissono L., Goy A., Petrone G., Segnan M., Torasso P., Intrigue: personalized recommendation of tourist attractions for desktop and handset devices, Applied AI, Special Issue on Artificial Intelligence for Cultural Heritage and Digital Libraries 17 (8-9) (2003) 687–714.

[3] Burke R., The Adaptive Web, chap. Hybrid web recommender systems, Springer Berlin / Heidelberg, 2007, pp. 377–408.

[4] Delgado J., Davidson R., Knowledge bases and user profiling in travel and hospitality recommender systems, in: Proceedings of the Enter Conference, Springer Verlag, 2002.

[5] Felfernig A., Gordea S., Jannach D., Teppan E., Zanker M., A short survey of recommendation technologies in travel and tourism, OEGAI Journal 25 (7) (2007) 17–22.

[6] Fesenmaier D.R., Ricci F., Schaumlechner E., Wober K., Zanella C., Dietorecs: Travel advisory for multiple decision styles, in: Information and Communication Technologies in Tourism, 2003.

[7] Fink J., Kobsa A., User modeling for personalized city tours, Artificial Intelligence Review 18 (1) (2002) 33–74.

[8] Garcia I., Sebastia L., Onaindia E., A negotiation approach for group recommendation, in: The International Conference on Artificial Intelligence (IC-AI), 2009.

[9] Hanani U., Shapira B., Shoval P., Information filtering: Overview of issues, research and systems, User Modeling and User-Adapted Interaction 11.

[10] Jameson A., More than the sum of its members: Challenges for group recommender systems, in: Proceedings of the International Working Conference on Advanced Visual Interfaces, 2004.

[11] Jameson A., Baldes S., Kleinbauer T., Enhancing mutual awareness in group recommender systems, in: B. Mobasher, e. Sarabjot S. Anand (eds.), Proceedings of the IJCAI, 2003.

[12] Jameson A., Baldes S., Kleinbauer T., Two methods for enhancing mutual awareness in a group recommender system, in: Proceedings of the International Working Conference on Advanced Visual Interfaces, 2004.

[13] McCarthy K., McGinty L., Smyth B., Salam M., Social interaction in the cats group recommender, in: Workshop on the Social Navigation and Community based Adaptation Technologies, 2006.

[14] Pazzani M.J., A framework for collaborative, content-based and demographic filtering, Artificial Intelligence Review 13 (1999) 393–408.

[15] Plua, C. and Jameson, A., Collaborative preference elicitation in a group travel recommender system, in: Proceedings of the AH 2002 Workshop on Recommendation and Personalization in eCommerce, 2002.

[16] Ricci F., Travel recommender systems, IEEE Intelligent Systems (2002) 55–57.

[17] Ricci F., Arslan B., Mirzadeh N., Venturini A., Itr: A case-based travel advisory system, in: Proceedings of the 6th European Conference on Advances in Case-Based Reasoning, Lecture Notes In Computer Science, vol. 2416, 2002.

[18] Ricci F., Cavada D., Nguyen Q.N., Integrating travel planning and on-tour support in a case-based recommender system, in: Proceedings of the Workshop on Mobile Tourism Systems (in conjunction with Mobile HCI'02), 2002.

[19] Sebastia L., Garcia I., Onaindia E., Guzman C., e-Tourism: a tourist recommendation and planning application, in: 20th IEEE International conference on Tools with Artificial Intelligence (ICTAI), 2008.

[20] Sebastia L., Garcia I., Onaindia E., Guzman C., e-Tourism: a tourist recommendation and planning application, International Journal on Artificial Intelligence Tools (IJAIT). In press.

[21] Venturini, A. and Ricci, F., Applying trip@dvice recommendation technology to www.visiteurope.com, in: ECAI, 2006.

[22] Werthner H., Intelligent systems in travel and tourism, in: Proceeding of the 18th International Joint Conference on Artificial Intelligence IJCAI, 2003.