Preference Elicitation and Negotiation in a Group Recommender System

Jesús Omar Álvarez Márquez[(✉)] and Jürgen Ziegler

University of Duisburg-Essen, Duisburg, Germany
{jesus.alvarez-marquez,juergen.ziegler}@uni-due.de

Abstract. We present a novel approach to group recommender systems that better takes into account the social interaction in a group when formulating, discussing and negotiating the features of the item to be jointly selected. Our approach provides discussion support in a collaborative preference elicitation and negotiation process. Individual preferences are continuously aggregated and immediate feedback of the resulting recommendations is provided. We also support the last stage in the decision process when users collectively select the final item from the recommendation set. The prototype hotel recommender Hootle is developed following these concepts and tested in a user study. The results indicate a higher overall satisfaction with the system as well as a higher perceived recommendation quality when compared against a system version where no negotiation was possible. However, they also indicate that the negotiation-based approach may be more suitable for smaller groups, an aspect that will require further research.

Keywords: Group recommender system · Group preference elicitation · Negotiation · Decision making

1 Introduction

Over the recent years, recommender systems have proven beneficial in supporting users when selecting or buying items from large sets of alternatives [30]. Buying something in a virtual shop, deciding which film to watch or planning where to go on holidays can easily become a tedious task when solely relying on manual search and filtering techniques, which may lead to information overload and choice difficulties. Therefore, the importance of recommender systems has increased fast in the last years, being now used widely throughout the internet. While the field of recommendations for single users has already been deeply explored, the same cannot be said about group recommender systems. Even though a significant number of group recommenders have been developed in the past years [5, 18], there is still a range of issues which have not been sufficiently investigated so far.

Most group recommending approaches rely on existing user profiles which are either aggregated into a single group profile (model aggregation) before generating group recommendations, or which are used for calculating individual recommendations that are subsequently aggregated, using a variety of different strategies (recommendation aggregation). However, while sufficient profile information is often not available

in the case of single users – either due to a cold start condition, or because users do not want their profile to be stored – this problem is even more pertinent for groups where the likelihood of each user having a stored profile that can be exploited by the recommender is relatively low. This is especially the case for ad hoc groups who gather spontaneously or who come from different organizational contexts. A further issue is the situational variability of the group members' preferences. This is also a problem in single-user recommending, but is aggravated by the fact that the inherent heterogeneity of preferences in a group may be amplified due to different responses to the situational context. These issues ask for methods that can elicit group preferences on the fly and that can aggregate individual preferences in a manner that best suits the individual users as well as the group as a whole.

Solving the complex trade-off between the degree of satisfaction of individual users and the group as a whole is typically attempted by applying one out of set of fixed strategies, such as averaging the satisfaction of all group members or minimizing discomfort for the least satisfied user. However, fixed strategies do not take the dynamics of group settings and situational needs into account. In particular, the aspect of social interaction when moving towards a joint decision is typically not sufficiently supported in existing group recommenders.

In this paper, we propose a novel method that tries to approach group recommendations from the point of intersection of traditional group recommenders and group decision making theory, allowing users to collaboratively create a preference model (thus addressing collaborative preference elicitation [28]), from which recommendations are generated. In this process, group interaction can happen at two (tightly intertwined) stages: (1) users can online discuss and negotiate preferences stated by others, and (2) they can discuss and rate items taken from the recommendation set to arrive at a final consensus decision.

Following the idea that computer-mediated discussion groups have more equal member participation [32], the goal is to avoid unfair situations in which some users might not be satisfied with the items proposed by the system. Our system supports remote online negotiation, although the approach can also be adapted to co-located settings. Each user can specify an individual preference model by freely adding desired features, using an explicit preference elicitation approach [27]. The individual preferences are then aggregated to form the group preference model and to determine an initial set of recommendations. All members' preferences, as well as the group aggregation, are visible to the participants. Most importantly, individual preferences can then be negotiated in a system-supported manner: by group discussion, members may thus be able to convince other users to modify their preferences, so the group model changes to better match all members' desires. Recommendations are continuously calculated and updated when the group preferences change, thus allowing users to immediately see the effect of their actions. Different mechanisms are provided for discussing and reaching an agreement, both for the creation of a group preference model and for the final item selection.

In the following, we first survey related research before presenting the conceptual aspects of our approach. We then describe the prototype implementation *Hootle* and its user interface design. We report on a user study we performed with groups of different sizes and conclude by summarizing our work and outlining future work.

2   Related Work

While the field of recommending items for single users has already received a great deal of attention in recent research, leading to quite effective recommendation methods, recommender systems for groups are, in comparison, a still less deeply investigated area. Various group recommender systems have been developed over the recent years, starting from early systems such as *MusicFX* [19], a group music recommender, that use different approaches for generating recommendations [5, 12]. However, there are still many open research questions concerning, for example, the best approach to aggregating individual preferences, techniques for responding to the situational needs of the group, or supporting the social interaction processes in the group for converging on a joint decision.

To structure the wide range of different aspects involved in group recommending, [14] suggest a design space comprising the dimensions preference input (including dynamic aspects), process characteristics, group characteristics, and (presentation of) output. In the process dimension, an important aspect is how individual, possibly conflicting preferences can be merged to obtain recommendations that best fit the group as a whole. Although different approaches in group recommenders gather and represent users' preferences in different ways, they commonly use one of two schemas [12]:

Aggregation of Item Predictions for Individual Users (Prediction Aggregation). This approach assumes that for each item, it is possible to calculate a user's satisfaction, given the user's profile. Then, using the calculated predictions and making use of some specific aggregation strategy, items are sorted by the group's overall satisfaction. In [9] a video recommender that uses this strategy is described; also *Polylens* [26], a system that suggests movies to small groups of people with similar interests, based on the personal five-star scale ratings from *Movielens* [8] uses this method.

Construction of Group Preference Models (Model Aggregation). Instead of predicting matching items for each user, the system uses information about individual members to create a preference model for the group as a whole. Recommendations are generated by determining those items that best match the group model. The number of possible methods for creating the group's model is even bigger than it is for prediction aggregation strategies. For example, in *Let's Browse* [15] the group preference model can be seen as an aggregation of individual preference models. In *Intrigue* [1, 2] (which recommends sightseeing destinations for heterogeneous groups of tourists) the group preference model is constructed by aggregating preference models of homogeneous subgroups within the main group. *MusicFX* [19] chooses background music in a fitness center to accommodate members' preferences, also by merging their individual models. AGReMo [4] recommends movies to watch in cinemas close to a location for ad hoc groups of users, creating the group's preference model not only by individual model aggregation but also taking into account some specific group variables (e.g. time, weight of each member's vote). Furthermore, the *Travel Decision Forum* [10, 11] creates a group preference model that can be discussed and modified by the members themselves, aiming to non-collocated groups who are not able to meet face to face, allowing asynchronous communication.

Regardless of whether the aggregation is made before or after generating recommendations, an aggregation method that is appropriate for the specific group characteristics needs to be chosen. There are a number of voting strategies, empirically evaluated in [18], that have been used in actual group recommender systems. Some typical strategies (and systems using it) are:

– **Average strategy**, where the group score for an item is the average rating over all individuals (*Intrigue, Travel Decision Forum*).
– **Least misery strategy**, which scores items depending on the minimal rating it has among group members (*Polylens, AGReMo*).
– **Average without misery strategy**, consisting in rating items using an average function, but discarding those where the user score is under a threshold (*MusicFX, CATS* [20–23]).
– **Median strategy**, which uses the middle value of the group members' ratings (*Travel Decision Forum*).

On another dimension, the question of preference elicitation has to be solved, which is concerned with how the user-specific preference information needed to generate recommendations is obtained. One approach is to let users rate a number of items in advance and to derive preferences from this set of ratings. *AGReMo*, for instance, requires group members to create their own model of individual preferences before the group meeting takes place by rating movies that they already saw. In *Travel Decision Forum* each participant starts with an empty preference form that has to be filled with the desired options, so group members define new preferences for each session. A more interactive approach, although for single user systems, is described in [17] which requires users to repeatedly choose between sets of sample items that are selected based on latent factors of a rating matrix. The techniques mentioned also address the cold-start problem when no user profile is available up-front but initially require some effort on the part of the user to develop a sufficiently detailed profile.

However, most preference elicitation techniques do not take group interaction into account. As pointed out in [16], to obtain adequate group recommendations it is not only necessary to model users' individual preferences, but also to understand how a decision among group members is reached. While research on group decision making [31] is concerned with collaboratively making choices, focusing on the social process and the outcome, these aspects have mostly not been addressed in the development of group recommender systems. The process of group decision making involves a variety of aspects, such as the discussion and evaluation of others' ideas, conflict resolution, and evaluating the different options that have been elaborated. Also interesting for our research is the concept of consensus decision-making [7], which seeks for an acceptable resolution for the whole group. Within this context, Group Decision Support Systems (GDSS) have emerged, that aim at supporting the various aspects of decision making [24, 25]. Only few recommender systems attempt to include aspects of group decision theory, for instance, by introducing automated negotiation agents that simulate discussions between members to generate group recommendations [3]. However, supporting the entire preference elicitation and negotiation process that may occur when users take recommender-supported decisions is, to our knowledge, not realized by current group recommenders.

Lastly, taking into account the social factor that is involved in group recommendation, one needs to contemplate the question whether a user would be willing to change personal preferences in favor of the group's desires, bringing up the importance of group negotiation. Again in the *Travel Decision Forum*, users are able to explore other members' preferences, with the possibility to copy them or propose modifications. The *Collaborative Advisory Travel System* (*CATS)* focuses on collocated groups of persons gathered around a multi-touch table. Recommendations are made by collecting critiques (users' feedbacks respecting recommended destinations) that can be discussed face to face, since the system gives visual support to enhance awareness of each other's preferences. The main difference between CATS and the system proposed here is that the former is focused in critiquing items once they have been recommended, while the latter allows negotiation already in the preference elicitation stage.

3   Preference Elicitation and Negotiation Method

The method developed involves an iterative process of specifying, discussing and negotiating preferences in a remote collaboration setting. Instead of only discussing recommendations produced based on user profiles, interaction among group members is supported right from the beginning of the preference elicitation process. The overall process comprises the following stages which are not meant as sequential steps but which can basically be performed in any order (algorithmic and interface details are described in the next chapter):

1. Users begin by selecting desired features from a set of attributes describing the items available. Since the feature sets may be very large (e.g. cities in our example hotel recommender, users can first search for the features they want and place them in a private area).
2. By moving a feature to the user's individual preference list, the feature becomes active and is visible to other group members. Several features can be placed and rank ordered according to the relevance they have for the user.
3. The individual feature lists are constantly aggregated in a common, ranked group preference list and the recommendations that best match the current group model are immediately generated and shown to the group.
4. Users can discuss preferences stated by others and negotiate them by using a 'petition' function, potentially trading in own preferences for features other users want. Based on the discussions and negotiations, users may change their preferences which is again immediately reflected in the group model and the resulting recommendations.
5. From the recommendations users can at any time select the item(s) they really like and propose them to the other participants who can accept them or propose alternatives. Also in this stage of the process, discussions are supported by the system.

The closed loop interaction with immediate feedback in the group model and the recommendations increases participants' awareness of others' preferences and the effects their own preference changes have on the group results. The approach also entails aspects of critique-based recommenders since users can criticize or accept

proposed features or recommended items. In contrast to fully automated recommender system, users have a higher level of control over the process and can easily adapt it to their current situational needs and context.

4   Description of the System

To demonstrate our approach we designed and implemented a prototype group recommender system that employs content-based techniques. The system is in principle applicable in a wide range of application areas, such as candidate selection, requirements specification, or leisure activities, as long as it is possible to obtain the properties of the items to be recommended. For demonstration purposes, we chose hotel selection for group travel as application area and use an Expedia dataset consisting of 151.000 hotel entries with descriptive information.

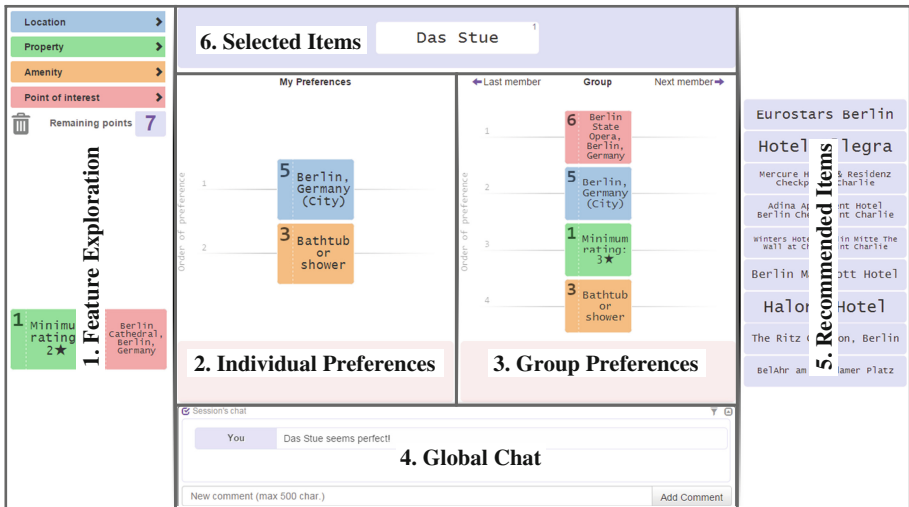Figure 1 shows a screenshot of the user interface, described as following:

Fig. 1. Areas of the interface.

1. **Feature exploration.** This area consists of a set of defined filters that let users search for specific attributes and a space to store the selected ones. For example, filters could be location, facilities or nearby points of interest.
2. **Individual preferences.** Features selected in area 1 can be added here by drag-and-drop, meaning that the user wants these features to be present (or excluded) in the recommended items (more details in the section about Individual Preferences). Users can also rank their preferences to express different levels of importance.

3. **Group preferences**. A ranked aggregation of all individual preferences is displayed in this area. It is also possible for users to navigate through the preferences of other participants here.
4. **Global chat.** In this section, the group can discuss arbitrary questions that come up in the decision process. Requests for preference changes ("petitions") and comments about specific features can also be displayed here.
5. **Recommended items.** Here, the items that best match the current group preferences and their relative weight are shown. The list is constantly updated in real-time when users add or change features.
6. **Recommendations selected by users.** From the recommendations area, users can pick the items they like most, and place them here. This space works as a shared area, so each item added here is visible to all participants.

4.1   Feature-Based Preference Elicitation

Individual preferences are defined by each group member by selecting features from the exploration area, where they can use different filters to locate them. Later, features can be placed into the user's individual preference space. The system allows to specify both positive and negative features.

Positive features. Apositive property means that a user wants it to be found in the recommendations. Users can specify an order of preference among positive attributes by dragging them to a higher or lower position in the list, which denotes the degree of importance that the user gives to each feature. Multiple features may have the same preference level.

Negative features. Negative properties are those that the user does not want to get as feature of the recommended items. They are placed inside a subspace within the
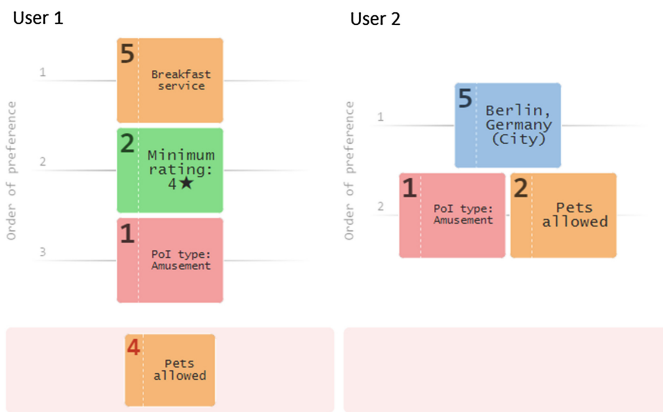
Fig. 2. Example of preference areas belonging to two different users. The ordered list represents the positive (desired) attributes, while the area at the bottom contains the negative (vetoed) ones. The cost of each attribute can be found at the top-left corner.

individual area (Fig. 2), called the veto area. Vetoed attributes have no preference order.

Cost of features. When users specify a large number of features as preferences, several problems may arise: first, it may be difficult to create meaningful integrated group preferences because the probability that features contradict each other increases, requiring more complex and longer negotiation processes. Second, users may over-specify their preferences making it difficult or impossible to calculate well-matching recommendations. We therefore decided to devise a mechanism that gently pushes users towards only specifying the features they really want.

For this purpose, a method for measuring the cost of each feature has been implemented. Each attribute has a related cost depending on how restrictive it is (i.e. how many items are left after using it as filter over the database). When a user selects a feature he or she pays for it from a limited budget. Users only have a number of tokens to exchange for attributes so they have to choose which ones are most important. This way, users selecting very restrictive features will only be able to create a small list of preferences as they will cost more tokens. It is also necessary to remark that the cost for positive attributes differs from the one for negatives. Positive attributes are more expensive the more restrictive they are; for negative features, more restrictiveness means less cost.

Group Preferences. While creating their individual preference lists, users can immediately see the overall results for the group. Inside the group preference area, an aggregation of all individual user preferences is displayed. This list is called the group preference list. The aggregation of individual preferences is performed using a variant of the Borda Count method, combined with rules regarding the vetoed attributes.

Borda Count is a voting method in which voters rank options or candidates in order of preference. In standard Borda Count, each option receives a score depending on its rank, and to obtain the aggregated score the points that each voter has given to it are summed up. In the case at hand, not only the rank of each option has been taken into account, but also its cost. When a user chooses to place a relatively expensive (restrictive) feature in the individual preference list, it is fair to think that the user cares more about this specific attribute. The equation used to calculate the aggregated score of an attribute *i* is presented in (1), where u is the number of group's members, n is the total number of different attributes used, p_{ij} is the preference value given to the attribute i by the user j, c_i is the cost of the attribute i and λ is used to correct the importance of the cost (with $\lambda = 0$ the result would be a standard Borda Count voting aggregation).

$$PAtt_i = \sum_{j=0..u} \left(\frac{1}{n} (n - p_{ij}) \right) + \frac{\lambda c_i}{n} \tag{1}$$

Attributes only receive points if users include them in their preferences. Finally, the group preference list is created by calculating the total score for each item and sorting them as usual (Fig. 3).

Vetoing a feature is a strong statement, it means that the person who stated it really does not want items with this feature. It would be desirable to avoid this feature, even if
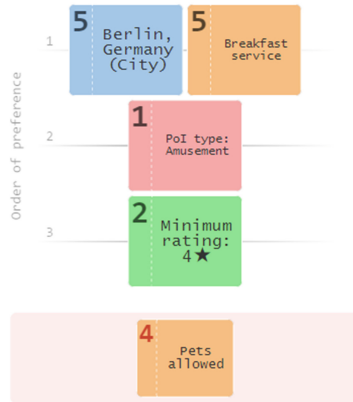
Fig. 3. Resulting preference setting for the group, using the individual lists shown at Fig. 2.

someone else in the group still wants it. Thus, vetoed attributes are removed from the group preference list and will not appear in any of the recommendations.

4.2   Generating Recommendations

Based on the aggregated user preferences the system applies a content-based filtering method to generate recommendations (Fig. 4). In content-based filtering, items are described by a set of attributes, and each user has a profile of preferences indicating the item properties the user likes. In our case, the individual preference set in a session represents the full user profile, thus, the system is applicable in cold-start situations where no user profile exists yet.

To generate recommendations, group preferences are compared to the items' properties in order to find the best matching ones. First, the system removes all the items
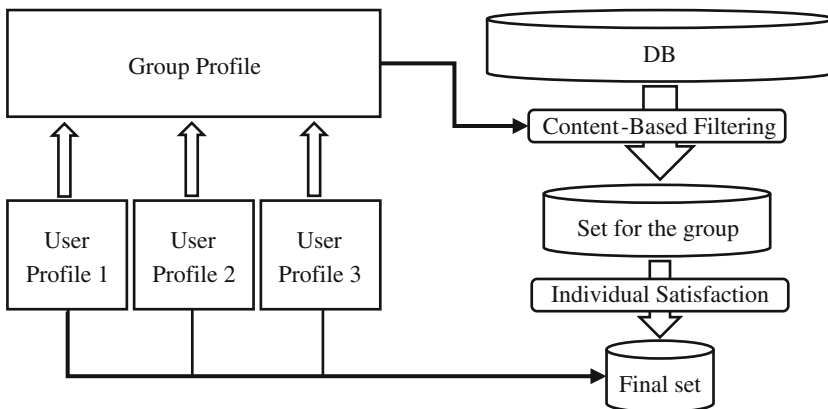

Fig. 4. Scheme of the filtering process.

that contain a vetoed attribute. The remaining items receive a score based on how many positive features they match, their total score being the sum of their attributes' values. The value of each attribute comes defined by the Borda Count method previously described, so attributes with higher preference levels will give higher score values to the items containing them. For distance attributes (coordinates, regions or points of interest), the value they were assigned by the Borda Count is modified depending on how far an item is from the given feature (closer items obtain higher scores).

If the system would simply present the ten top scored items, it could happen that for some users whose attributes are lower in the group preference list, no good options are returned. Since the main purpose of the system is to provide a negotiation environment, it seems necessary to return a well-balanced set of items, in terms of member satisfaction. For this reason, a subset of items is extracted, within the already found, in a way that for each user there is at least one acceptable option, but giving at the same time importance to the items that satisfy the group as a whole. An item is considered acceptable for a participant when his/her satisfaction level concerning this option is higher than a given threshold. Satisfaction is calculated taking into account the individual preference model defined by a user, in a similar way an item's group score is calculated, but divided by the maximum points an item could receive (that is, when an item contains all the features a user wants). Finally, the selected items are presented to participants in the recommendation area of the screen (5 - Recommended Items in Fig. 1).

As said before, the system is applicable without requiring the prior availability of stored user profiles which is particularly beneficial in group contexts for the reasons mentioned earlier. However, in principle more complex and longer-term user profiles could be built if past choices were saved for future sessions. If this option was used and is acceptable for users, the interaction effort needed for specifying the desired features could be reduced, just specifying changes in the existing profile, and possibly increasing the precision of the recommendations.

4.3   Negotiation

User preferences are typically not a static phenomenon but are influenced by the situational context of the group and the social interaction that takes place within it. Users may also differ in the extent to which they have already formed their objectives at the beginning of the group process. They may react to preferences expressed by others, either accepting or rejecting them. They may also be willing to dispense with a desired feature if someone else in the group accepts one of their other preferences, thus embarking on a negotiation process with other group members. For these reasons, our system provides several functions that specifically support discussion, negotiation and consensus finding among group members.

Communication. Users need the possibility to express their opinions about the decision process as a whole as well as about specific preferences stated by others. To support these types of communication, two methods are implemented in the system.

Discussion threads and global chat. Each feature has its own discussion thread, which means that users can access it and say what they think about a specific property, keeping the comments organized by attribute. A global chat is also available, placed in area 4 displayed in Fig. 1. The global chat lets participants talk about arbitrary aspects of the current session, and also informs group members about recent updates in specific comment threads.

Petitions. Petitions are requests such as removing a feature or changing its rank. It is not possible to request the addition of an attribute, as adding a feature to one's individual list is already an implicit petition to the rest of the group: every user wants the others to adopt the same preferences as he/she has, since this would increase the fit of the recommendations with this user's wishes.

Finding and Resolving Conflicts. Conflicts appear when two or more participants want features that contradict each other. Several mechanism help to resolve such situations. First, users can explore the individual preferences of other participants and discuss them if a conflict occurs.

Second, once a set of recommendations is presented, users can access information about each item recommended. Also, those entries in the group preference list that are not fulfilled by an item are highlighted in that list. Thus, when a user likes a recommendation, he/she can see the preferences that are in conflict with it and try to change the opinion of the members who added them.

Finally, for each recommendation, the calculated grade of satisfaction of each user can be displayed in a spider diagram, so the group may choose items that are more balanced with respect to the members' individual desires (i.e. are less conflictive).

Proposing Items. From the recommendation area, users are able to express their approval for a specific recommended item by placing it into the "recommendations selected by users" space (area 6 in Fig. 1). This step shows the group that one user likes a recommendation and proposes it as option. The other participants now can accept it as a good option, reject it or just ignore it, waiting for more proposals to show up.

4.4   Repeat and Decide

The *"adding features-get recommendations-negotiate"* cycle can be repeated several times, narrowing down the recommendations given with each new iteration, until the group reaches agreement. If and when consensus is reached, however, is something that only the group itself is able to decide. As has been said in the previous section, users can add items that they like into a shared area, so the others can express their acceptance about it. For some groups, the item to be finally selected may be the one that is accepted by more than fifty percent of the members; in other cases, there may be situations where all users have accepted an item except one who finds it unsatisfactory. While a fixed group recommendation strategy, for example, a 'least misery' approach that might seem applicable in the latter case, would always try to satisfy user needs in one prescribed manner, we believe that the system cannot generally resolve such decision problems. Although the system provides tools for preference specification,

discussion and acceptance measuring, it is up to the users to decide whether a recommendation fits their needs or not and to make the final choice.

5   Evaluation

To evaluate our approach, we performed a user study with several groups comprising between three and five users. We did not consider larger groups at this point because we believe this group size to be typical for the application domain chosen which is selecting a hotel for a joint leisure or business trip. Also, Hootle, our Web-based prototype implementation of the approach, while still work in progress, is stable enough to support this group size but still has to be tested for larger-scale trials. The main objectives of this study were to determine the usability of the approach and the quality of the resulting recommendations, as well as, more specifically, to analyze the impact of the cooperative preference elicitation and negotiation tools developed.

5.1   Setting and Experimental Tasks

To assess whether the preference elicitation, negotiation and recommendation methods developed benefit group decision processes, we tested two different versions of the system where one served as baseline for comparison. While one system version provided the full set of functions described including group discussion support (hereafter version D – Discussion), we restricted the second version to specifying preferences and calculating recommendations (version ND – No Discussion), similar to a conventional group recommender system, but still offering the possibility to specify preferences in an ad hoc manner without using existing user profiles. We decided against using an existing alternative group recommender for comparison because the systems would have differed in too many aspects, making it difficult to pinpoint the specific benefits of the proposed innovations. In both cases, we make use of a hotel database provided by Expedia with 151,000 entries. For each hotel, a full description and a set of attributes, including property and room amenities (within a total of 360 possibilities), locations (258,426) and points of interest nearby (94,512) was available. We deliberately decided to focus the negotiation and decision process on the objective properties of the items, excluding price information which would have opened up additional questions concerning economic concerns and behavior in the test groups. This aspect, however, will be subject of future research.

We prepared two types of task scenarios with different levels of complexity:

- In an 'introductory' task, the group was instructed to select a hotel knowing beforehand some common, desired attributes, as well as the location of the hotel. This task also served as a training session for the application, to allow participants to explore the functions and possibilities the system supplies. Two scenarios for this task were presented:
  - Your group will be participating at a conference in Berlin. As the conference always provides lunch and dinner, you just need to find a hotel including breakfast. Your conference takes place near the Brandenburg Gate.

– Your group wants to enjoy some days on the beach. You already decided to go to an apartment, as you want to prepare meals on your own. Everyone loves Spain so you also decided to go to Marbella.

- In the 'open' task which was always performed after the introductory task, only unspecific instructions were given to the group such as "Find a place to stay during summer vacation". The possible scenarios were:
  – It is summertime. You and your friends really need to get out of the daily routine. Discuss where to stay.
  – Your group wants to do some kind of city trip. Where are you going to?

To avoid the problem that in a test situation, participants do not bring with them the objectives and preferences they would have in a real-life decision situation, or might comply too quickly with the wishes of other participants, we tried to artificially induce different backgrounds and objectives for each group member. For this purpose, we created a set of role cards for the second task, depending on the scenario used. With this method, we expected to generate conflicts and discussion when randomly distributing the role cards among group's members. As an example, the role cards for the first scenario in task 2 were (abbreviated here):

1. You're a sport addict. You like to eat healthy and don't trust in hotel food. You hate giant hotels and prefer small pensions or camping sites.
2. You're allergic to nearly everything. Vacation at a camping site would be like a death sentence to you. You prefer the pool over the sea. You don't want to do anything so you prefer all inclusive.
3. You like to go for long hikes. You're fascinated by mountains. You don't want to cook but you won't be there during the day so you just need breakfast and dinner.
4. You're into cultural things. If you go on vacation, you want to see things. You also like to go out for dinner so breakfast only would totally fit your needs.
5. You like to party. As you won't be able to prepare your own food, there should be someone who helps you with this. More important is the location of your hotel. Nobody wants to walk for an eternity to go clubbing.

5.2   Method

A total of 48 students were recruited as participants (5 male, 43 female, average age of 20.94, σ 5.018), distributed in groups of different sizes: 4 groups of 3 persons (12), 4 groups of 4 persons (16) and 4 groups of 5 persons (20). Two groups of each size ran a full version of the system (D), while the other two groups tested the version without negotiation support (ND). Since the system is Web-based, all users were provided with a normal desktop computer with a display screen of 21 in and running the same browser. They sat in a large lab room but were separated from each other and instructed to only communicate via the means provided by the system.

Each group first received a brief introduction to the system which was dependent on whether the negotiation support was turned on or off for the group. After a brief trial, they were asked to work on the two decision tasks, always in the order introductory

task – open task. Before beginning the second task, they all received randomly one of the role cards.

For the groups using version D, a task was considered complete when they reached consensus about their preferred hotel or when they decided that it was not possible to find agreement. Since the groups with version ND were not able to communicate at all, their job consisted in defining their own preference model and, when the whole group had done this, each user separately selected a hotel from the resulting set of recommendations.

The first task including the explanation of the system was limited to a maximum of 40 min. As the explanation was no longer necessary, the second task, although more complex, should also be completed during this time.

After completing both tasks, participants were asked to fill in a questionnaire regarding aspects such as the quality of the recommendations or the ease-of-use of the system, using a 1-5 scale. The questionnaire comprised the SUS items [6] to compare the system against a well-established baseline as well as items from two recommender-specific assessment instruments (User experience of recommender systems [13] and *ResQue* [29]). The recommender-specific items were measuring mainly the constructs *user-perceived recommendation quality*, *perceived system effectiveness, interface adequacy,* and *ease of use*.

5.3 Results and Discussion

All tasks were finished within the allotted time. The D an ND groups differ on a considerable number of criteria. The members in ND groups were not able to choose the same hotel in a single instance. In two of these cases, some users couldn't even find a hotel that they liked when realizing the open task. On the other hand, all groups with version D were able to choose one unique hotel in both tasks, despite starting the

Table 1. Results of the questionnaire (all the D/ND differences p > 0.05, effects of group size were significant).

System version		No discussion				Discussion			
Group Size		3	4	5	Avg.	3	4	5	Avg.
Overall	m	3.40	3.00	3.70	3.39	4.33	4.00	3.60	3.92
satisfaction	σ	0.54	1.20	0.48	0.83	0.51	0.53	0.96	0.77
Would	m	3.20	2.38	3.30	2.96	3.50	3.25	3.30	3.33
recommend it	σ	1.30	1.06	0.67	1.02	0.83	0.70	1.06	0.86
Would use it again	m	2.40	2.50	3.10	2.74	3.17	3.13	3.00	3.08
	σ	0.89	0.92	1.10	1.01	0.75	0.99	0.66	0.77
Would use it	m	1.60	1.88	2.30	2.00	2.67	2.75	2.70	2.71
frequently	σ	0.54	0.64	0.67	0.67	0.81	1.04	0.94	0.90
Recommendations	m	3.20	3.38	3.80	3.52	4.33	3.38	4.00	3.88
were well chosen	σ	0.83	0.74	0.78	0.79	0.51	0.74	0.47	0.68

process with strongly different individual preferences. To achieve this joint decision, users had to iterate several times through the *"adding features-get recommendations-negotiate"* cycle, as well as to renounce some desired features due to the influence exerted by other members through discussions and petitions.

In terms of overall usability, both system versions received a SUS score which can be considered as borderline good with no differences between the two systems (ND = 68, D = 69). We performed a 2 × 3 ANOVA with system version and group size as independent variables and questionnaire item scores as dependent variables. Most item responses did not show significant differences between the two system versions which may be due to the limited number of groups tested. In Table 1, we list some of the results that were significant at a .05 level. Users in the discussion condition were overall more satisfied with the system, are more likely to recommend it to others and would be willing to use the system again and also more frequently. Also, the accuracy of the recommendations was rated higher in the discussion groups. While these results speak in favour of the discussion version, there appears to be an interesting interaction effect between system versions and group size. Generally, satisfaction and willingness to use and recommend the system tend to be higher for the small groups than the large groups when discussion is available. Concerning recommendation quality, the largest group had the highest ratings in the no-discussion condition while this is reversed in the discussion condition where the smallest group had the highest rating. This picture is somewhat blurred by the fact that the medium-sized groups (4 persons) had the largest variability so there is no clear relation between group size and these variables.

For the remaining questionnaire items (which we cannot report here fully due to space limitations) there is a tendency in favour of the discussion version both in the items related to usability and acceptance of the system as well as concerning the fit of the recommendations and the ease with which a matching hotel could be found.

The time needed to come to a decision differed significantly between the introductory task and open task (13,500 vs. 26,333, p = 0.05). Results concerning negotiation behavior are listed in Table 2: both individual changes and number of petitions increase with group size. In relation with Table 1, it may be concluded that users in small groups are generally more satisfied because they were able to select more preferences for themselves and made less changes in their individual lists (keeping their initial wishes).

Discussion: The results of this study can only give a first indication of how well the proposed approach works in comparison to other techniques and in different group

Table 2. Objective results (lower and upper bounds at 95 % confidence interval).

	3 Participants			4 Participants			5 Participants		
	m	LB	UB	m	LB	UB	m	LB	UB
Time	21	31	10,9	17,2	7,23	27,2	21,5	11,4	31,5
Pref. Sel./Part.	3	2,37	3,62	2,31	1,76	2,85	2,80	2,31	3,28
Ind. Changes/Part.	7,33	1,79	12,8	10,1	4,64	15,7	13,1	7,61	18,6
Petitions/Part.	0,66	0	1,65	0,68	0	1,54	1,95	1,18	2,71

contexts. We can see significant advantages for our approach of including discussion and negotiation features in a group recommender in some relevant items, as well as a tendency in favour of the system in the majority of other items. However, it appears that the system may be more useful in small groups. This may be due to several factors: first, as larger groups require more communication and negotiation to obtain an acceptable end results, this may increase the complexity of the task and the interaction effort. This may be true for other group decision making systems as well but will require further research. A second factor may be artificially created by the experimental method used. Since users were instructed to play the roles described in their respective role cards, the diversity of preferences increased with group size, possibly making it more difficult to make sense of the diverse standpoints and to lead the negotiation towards a joint group decision. This may not be the case in typical real world settings where group members' viewpoints may be more homogenous due to the prior history of the group. Also, the role card method can only be taken as an approximation of a real situation. In any case, the observed tendencies raise interesting general questions concerning test scenarios for evaluating group recommender systems.

6   Conclusions and Outlook

We have presented a novel approach to group recommending that provides more interactive control over the recommendation process than typical group recommenders and that does not require the prior availability of the group members' preference profiles, taking into consideration cold-start situations and potential privacy concerns. Most importantly, the method provides discussion and negotiation support in a collaborative preference elicitation and negotiation process. Individual preferences are aggregated in a group preference profile which is immediately updated when users change preferred features or their relevance level. Also, the resulting recommendations are continuously recalculated when group preferences change, and are always visible to the whole group. Since producing recommendations constitutes just an intermediate step in the group decision process, we also support group interaction in the final decision steps where the group needs to find consensus about the item finally selected.

The proposed technique provides much higher flexibility and responsiveness to situational needs than the fixed strategies typically used in group recommenders. While this research has focused on specifying preferences in an ad hoc fashion, the method can easily be extended by storing and re-using user profiles, thus reducing interaction effort to simply adapting an existing profile. Since the preferences of other users and resulting group preferences as well as the recommendations that match this profile are always visible, participants' awareness of individual and group views and of the effects of their preference settings is increased.

Based on these concepts, we developed the prototype hotel recommender Hootle and tested it in a user study. The results indicate a higher overall satisfaction with the system as well as a higher perceived recommendation quality when compared against a system version where no discussion was possible. However, we also saw an indication of an interaction effect between group size and the two system versions which suggests that the negotiation-based approach may be more suitable for smaller groups. Whether

this effect is due to the increased communication effort in larger groups, or may be dependent on the experimental scenarios used in the study is still an open question.

In future work, we aim at investigating the effects of group size more deeply and at optimizing the system to better scale for larger groups. A further work item is to consider alternative aggregation functions that may perform better than the Borda Count variant currently used. Finally, we aim at further improving the user experience with respect to the discussion and decision making features implemented. Also, more extensive empirical studies are planned, addressing also domains other than hotel selection.

References

1. Ardissono, L., Goy, A., Petrone, G., Segnan, M.: A multi-agent infrastructure for developing personalized web-based systems. ACM Trans. Internet Technol. (TOIT) **5**(1), 47–69 (2005)
2. Ardissono, L., Goy, A., Petrone, G., Segnan, M., Torasso, P.: Intrigue: personalized recommendation of tourist attractions for desktop and hand held devices. Appl. Artif. Intell. **17**(8–9), 687–714 (2003)
3. Bekkerman, P., Kraus, S., Ricci, F.: Applying cooperative negotiation methodology to group recommendation problem. In: Proceedings of Workshop on Recommender Systems in 17th European Conference on Artificial Intelligence (ECAI 2006), pp. 72–75. Citeseer (2006)
4. Beckmann, C., Gross, T.: Towards a group recommender process model for ad-hoc groups and on-demand recommendations. In: Proceedings of the 16th ACM International Conference on Supporting Group Work, pp. 329–330. ACM (2010)
5. Boratto, L., Carta, S.: State-of-the-art in group recommendation and new approaches for automatic identification of groups. In: Soro, Alessandro, Vargiu, Eloisa, Armano, Giuliano, Paddeu, Gavino (eds.) Information Retrieval and Mining in Distributed Environments. SCI, vol. 324, pp. 1–20. Springer, Heidelberg (2010)
6. Brooke, J.: SUS-A quick and dirty usability scale. Usability Eval. Ind. **189**, 194 (1996)
7. Hartnett, T.: Consensus-Oriented Decision-Making: the CODM Model for Facilitating Groups to Widespread Agreement. New Society Publishers, Gabriola Island (2011)
8. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. (TOIS) **22**(1), 5–53 (2004)
9. Hill, W., Stead, L., Rosenstein, M., Furnas, G.: Recommending and evaluating choices in a virtual community of use. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 194–201 (1995)
10. Jameson, A.: More than the sum of its members: challenges for group recommender systems. In: Proceedings of the Working Conference on Advanced Visual Interfaces, pp. 48–54. ACM (2004)
11. Jameson, A., Baldes, A., Kleinbauer, T.: Two methods for enhancing mutual awareness in a group recommender system. In: Proceedings of the Working Conference on Advanced Visual Interfaces, pp. 447–449. ACM (2004)
12. Jameson, A., Smyth, B.: Recommendation to groups. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 596–627. Springer, Heidelberg (2007)
13. Knijnenburg, B.P., Willemsen, M.C., Gantner, Z., Soncu, H., Newell, C.: Explaining the user experience of recommender systems. User Model. User-Adap. Inter. **22**(4–5), 441–504 (2012)
14. Kompan, M., Bielikova, M.: Group recommendations: survey and perspectives. Comput. Inform. **33**(2), 446–476 (2014)

15. Lieberman, H., Van Dyke, N., Vivacqua, A.: Let's browse: a collaborative browsing agent. Knowl.-Based Syst. **12**(8), 427–431 (1999)
16. Liu, X., Tian, Y., Ye, M., Lee, W.: Exploring personal impact for group recommendation. In: Proceedings of the 21st ACM International Conference on Information and Knowledge Management, pp. 674–683. ACM (2012)
17. Loepp, B., Hussein, T., Ziegler, J.: Choice-based preference elicitation for collaborative filtering recommender systems. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2014), pp. 3085–3094. ACM, New York (2014)
18. Masthoff, J.: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers, pp. 93–141. Personalized Digital Television. Springer, The Netherlands (2004)
19. McCarthy, J.F., Anagnost, T.D.: MusicFX: an arbiter of group preferences for computer supported collaborative workouts. In: Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work, pp. 363–372. ACM (1998)
20. McCarthy, K., McGinty, L., Smyth, B.: Case-based group recommendation: compromising for success. In: Weber, R.O., Richter, M.M. (eds.) ICCBR 2007. LNCS (LNAI), vol. 4626, pp. 299–313. Springer, Heidelberg (2007)
21. McCarthy, K., McGinty, L., Smyth, B., Salamó, M.: The needs of the many: a case-based group recommender system. In: Roth-Berghofer, T.R., Göker, M.H., Güvenir, H.A. (eds.) ECCBR 2006. LNCS (LNAI), vol. 4106, pp. 196–210. Springer, Heidelberg (2006)
22. McCarthy, K., McGinty, L., Smyth, B., Salamo, M.: Social interaction in the CATS group recommender. In: Workshop on the Social Navigation and Community Based Adaptation Technologies (2006)
23. McCarthy, K., Salamo, M., Coyle, L., McGinty, L., Smyth, B., Nixon, P.: CATS: A synchronous approach to collaborative group recommendation. In: FLAIRS Conference, vol. 2006, pp. 86–91 (2006)
24. McGrath, J.E., Berdahl, J.L.: Groups, technology, and time. In: Scott Tindale, R., et al. (eds.) Theory and Research on Small Groups, pp. 205–228. Springer, US (2002)
25. Nunamaker Jr., J.F., Briggs, R.O., Mittleman, D.D., Vogel, D.R., Balthazard, P.A.: Lessons from a dozen years of group support systems research: A discussion of lab and field findings. J. Manage. Inf. Syst. **13**, 163–207 (1996)
26. O'connor, M., Cosley, D., Konstan, J.A., Riedl, J.: Polylens: a recommender system for groups of users. In: Prinz, W., et al. (eds.) ECSCW 2001, pp. 199–218. Springer, The Netherlands (2001)
27. Pommeranz, A., Broekens, J., Wiggers, P., Brinkman, W.P., Jonker, C.M.: Designing interfaces for explicit preference elicitation: a user-centered investigation of preference representation and elicitation process. User Model. User-Adap. Inter. **22**(4–5), 357–397 (2012)
28. Pu, P., Chen, L.: User-involved preference elicitation for product search and recommender systems. AI Mag. **29**(4), 93 (2009)
29. Pu, P., Chen, L., Hu, R..: A user-centric evaluation framework for recommender systems. In: Proceedings of the Fifth ACM Conference on Recommender Systems, pp. 157–164. ACM (2011)
30. Ricci, F., Rokach, L., Shapira, B.: Introduction to Recommender Systems Handbook. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 1–35. Springer, US (2010)
31. Saaty, T.L.: Fundamentals of Decision Making and Priority Theory with the Analytic Hierarchy Process, vol. 6. RWS Publications, Pittsburgh (2000)
32. Walther, J.B.: Computer-mediated communication impersonal, interpersonal, and hyper personal interaction. Commun. Res. **23**(1), 3–43 (1996)