Generating Recommendations for Consensus Negotiation in Group Personalization Services

Maria Salamó · Kevin McCarthy · Barry Smyth

Abstract There are increasingly many personalization services in ubiquitous computing environments that involve a group of users rather than individuals. Ubiquitous commerce is one example of these environments. Ubiquitous commerce research is highly related to recommender systems that have the ability to provide even the most tentative shoppers with compelling and timely item suggestions. When the recommendations are made for a group of users, new challenges and issues arise to provide compelling item suggestions. One of the challenges a group recommender system must cope with is the potentially conflicting preferences of multiple users when selecting items for recommendation. In this paper we focus on how individual user models can be aggregated to reach a consensus on recommendations. We describe and evaluate nine different consensus strategies and analyze them to highlight the benefits of group recommendation using live-user preference data. Moreover, we show that the performance is significantly different among strategies.

Keywords Consensus · Group recommendation · Ubiquitous personalization services

M. Salamó
Dept. Matemàtica Aplicada i Anàlisi, Universitat de Barcelona, Gran via de les corts catalanes, 585 -08007 Barcelona, Spain
Tel.: +34-93-4039372
E-mail: maria.salamo@ub.edu

K. McCarthy
Clarity, School of Computer Science and Informatics, Science North, University College Dublin, Belfield, Dublin 4, Ireland
E-mail: kevin.mcarthy@ucd.ie

B. Smyth
Clarity, School of Computer Science and Informatics, Science North, University College Dublin, Belfield, Dublin 4, Ireland
E-mail: barry.smyth@ucd.ie

1 Introduction

Personalization services in ubiquitous environments — *ubiquitous personalization services computing*— has recently been emerging in many areas of society [25, 46, 47]. Ubiquitous commerce is one of these areas. In ubiquitous commerce, in order to support personalization services, a system should recognize preferred products (i.e., also known as items) through preference analysis such as a personal profile. Research on ubiquitous commerce has come up with a number of recommendation systems reaching out in the real world.

Recommender Systems are particularly well adapted to e-commerce applications where users need to be guided through complex product spaces. They combine user profiling and filtering techniques to provide more proactive and personal information retrieval systems, and have been gaining in popularity as a way of overcoming the ubiquitous information overload problem. Much of the research on ubiquitous recommendation systems are related to mobile phones and have been examined under the term m-commerce [49]. Examples include: a ubiquitous shopping system called *MyGROCER* [24], *MovieLens Unplugged* (MLU) [34], *Mobile Sales Assistant* [40], *Buying-net* [23], and *APriori* [38] product recommendation system. Apart from m-commerce applications, a great number of ubiquitous computing applications also apply recommendation systems. For example, there have been numerous works on recommending tourist attractions (e.g., COMPASS [43]) or more general recommendation algorithms for mobile devices [16].

While much of the research on ubiquitous recommendation systems has focused on making recommendations to the individual [23, 39], many scenarios involve groups of inter-related users [17, 41]. In these sce-

narios, typically a group of people are intending to participate in a group activity [35].

Recently some researchers have begun to look at the whole set of new challenges and issues associated with group recommendation [21]. It is no longer reasonable to generate a set of recommendations with respect to a single set of (relatively) consistent preferences, as is normally the case in the single-user scenario. Instead, groups of users are likely to introduce a diverse range of preferences where conflicting needs are likely to be commonplace and where different users may be more or less easy to satisfy [19].

The job of the group recommender is to make suggestions that reflect the preferences of the group as a whole, while offering reasonable and acceptable consensus options to individual group members. In this regard, a key aspect of group recommendation concerns the way in which individual users profiles are combined to reach a consensus on recommendation (or a set of recommendations) for the group at the completion of the group session.

The work presented in this paper continues previous work [30–32] on group recommendation. In particular, this previous work proposed specific group recommendation approaches and focused on interfacing functionality that assists individual group members in better understanding the evolving needs of the group. This helps group members to appreciate the compromises that may be required for a satisfactory conclusion to be reached (see also [20]). From previous work, one remaining unsolved issue is: how to reach a consensus when all users have completed interacting with the system and have chosen their preferred items? The aim of consensus negotiation is to help the group members arrive at a final recommendation which maximally satisfies the individual members preferences. In this paper we focus on the consensus-based recommendation technique, describing and evaluating nine different approaches for reaching a consensus on recommendations. Moreover, we describe the results of an evaluation, based on live-user data, as a means to explore the relationship between group diversity and consensus recommendation. It is worth mentioning that the proposed strategies start from the premise that each user has an individual user model that contains her preferences. They are defined accordingly to this premise with the aim of being general enough to be applicable to a wide range of group recommender scenarios. Notice that recommender systems can be distinguished by the type of feedback that they support; examples include *value elicitation, ratings-based feedback* and *preference-based feedback* [45]. In particular, we analyze the strategies in a conversational case-based reasoning recommender[1]. We use a form of user feedback called *critiquing* [5,33], where a user indicates a directional feature preference in relation to the current recommendation. To demonstrate that they are general enough, we compare our proposals with well-known state-of-the-art approaches normally used for recommendation generation in collaborative filtering.

Hence, the contributions of this work are two-fold. First of all, we propose a set of new strategies for reaching consensus. Second, we evaluate them based on live-user preference information, in which we compare the performance of different consensus strategies on different groups sizes and different types of user groups, ranging from groups with very similar preferences to more diverse groups with competing preferences.

The rest of this paper is organized as follows. In the next section we review related work on group recommender systems. Section 3 describes our proposed consensus-based recommendation strategies. Section 4 is devoted to a small example that help the reader to understand the proposals. Next, Section 5 discusses the experimental results. Finally, we conclude the paper with a discussion about the main results and implications of the work.

2 Related work

Ubiquitous computing research has uncovered many new situations where recommendation technology can have a critical role to play, and in many of these situations there may be more than one individual designated as the recipient of the recommendation. For example, the way in which we book a restaurant to go with our families using a mobile environment [36]. Accordingly a group of friends, or family members, now have access to hundreds of restaurants' content from which to choose. Recommendation technologies will help them to more effectively navigate through this content space, but in doing so must cater for a set of users rather than a single user. These types of scenario have motivated recent interest in group recommendation and to date a variety of early-stage systems have been developed in domains such as group web-page recommendation [26, 37,44], recommending vacations or tours to groups of tourists [1,19,31,43], recommending music tracks and playlists to large groups of listeners [8,29], and, recommending movies and TV programmes to friends and family [15,35,48]. Group recommenders can be distinguished, as defined in [21], according to their approach

---

[1] It is a form of content-based recommendation where individual items are described in terms of a well defined set of features.

to 4 basic recommendation sub-tasks —(1) preference elicitation; (2) recommendation generation; (3) presentation and explanation; (4) consensus negotiation.

Preference elicitation refers to the manner in which information is acquired from users and in many cases methods similar to those used in single-user recommender systems are applied. For example, preferences may be acquired by asking users directly (*explicit preference elicitation*) or by inferring their preferences from their actions and feedback (*implicit preference elicitation*). In the case of the former, systems such as the Travel Decision Advisor [19] and PolyLens [35] both acquire preferences by asking users to specify them explicitly; either in the form of preferred features or item ratings. In contrast, group systems such as FlyTrap [8] and Let's Browse [26] acquire the preferences by monitoring a user's interactions. FlyTrap, for example, learns about the preferences of individual users by mining each user's personal music usage habits. In ubiquitous computing explicit and implicit preference elicitation can be applied [3,39]. Most of the ubiquitous recommender systems use ratings, for example *Buying-net* [23]. In [22], apart from acquiring explicit preferences, the ubiquitous recommender additionally learns user preferences through learning from the user profile. In this work, the consensus-based proposals are devoted to be applicable to both: explicit and implicit preference elicitation.

Most of the research on group recommendation investigated the core algorithms used for *recommendation generation*. Two different strategies have been mostly used for generating group recommendations: aggregating individual predictions into group predictions (*aggregated predictions*) or aggregating individual models into group models (*aggregated models*). Differences among these strategies differ in the timing of data aggregation step. The *aggregated predictions* strategy [4,27] generated individual predictions based on individual preference models and then aggregates the individual predictions into a group prediction. The alternative strategy is to construct a group preference model prior to any item recommendation taking place [42]. The *aggregated models* strategy merges individual user models into a group-based model and then generates recommendations using the aggregated group model [31]. For example, GMK (Group Knowledge Management) [9] proposes a generic framework for management in context-aware group applications and services that extracts group knowledge regarding the involved users.

The final two subtasks (*presenting and explaining recommendations* and helping group members to *reach consensus*) have received less attention from researchers. That said, there is an increasing interest in these areas. This is particularly evident in group recommenda-

tion scenarios because convincing group members that a particular recommendation is right for them is especially important. For instance, Let's Browse [26] explains its Web page recommendations to group members by highlighting keywords from the page that are judged to be relevant to the group as a whole.

As highlighted in [21], many group recommenders do not explicitly support *consensus negotiation*. Very often it is assumed that one particular group member is responsible for the final decision; Let's Browse[26], makes this assumption, because one group member typically controls the system interaction with other group members playing the role of viewers rather than actors. Alternatively the role of the recommender could be to produce a set of recommendations that will ultimately be debated by the group offline, before a consensus is reached. In general, *consensus* remains an open issue for group recommenders. This paper proposes more active solutions to help users to reach consensus in a post-recommendation phase.

3 Consensus-based strategies

In this section, we propose different strategies to reach a consensus on the recommendations made for a group of users. First, we define some required definitions. Second, we present the consensus strategies divided into three groups: (1) *Statistical dispersion* strategies; (2) *Individual content* strategies; and (3) *Collaborative filtering* strategies. Note that statistical and content dispersion strategies are the proposals of this paper. Additionally, in this section we have also added the explanation of the state-of-the-art strategies commonly used in collaborative filtering algorithms for generating recommendations. We also consider how they can be applied when reaching consensus.

3.1 Required definitions

Let $P = \{p_1, ..., p_n\}$ be a set of products or items for recommendation, where p_i is the ith product. Depending on the recommender algorithm, P will be the whole set of products or a subset of them.

Let $U = \{u_1, ..., u_k\}$ be a set of users, where u_j represents the jth user and k the number of users, $|U| = k$. The set of individual user models will be defined as $IM = \{IM^{u_1}, ..., IM^{u_k}\}$, where IM^{u_j} represents the individual user model of the jth user. We consider that each user u_j has an individual user model $IM^{u_j} = \{I_1, ..., I_{r_{u_j}}\}$ that represents her individual preferences, from her initial preference I_1 to her last preference $I_{r_{u_j}}$. Note that the subindex r_{u_j} may be

different for each user in U because each user defines her own set of preferences. Thus, R is the total number of preferences for the group which is computed as $R = |IM^{u_1}| + ... + |IM^{u_k}| = r_{u_1} + ... + r_{u_k}$.

In subsequent sections we will use the satisfaction measure, as defined in Equation 1.

$$\delta(p_i, IM^{u_j}) = \sum_{s=1}^{r_{u_j}} w_s \cdot Satisfies(p_i, I_s) \tag{1}$$

The satisfaction measure, $\delta(p_i, IM^{u_j})$, shows for a particular product p_i and a user u_j, the number of preferences from her individual user model IM^{u_j} that product p_i satisfies. The parameter w_s is a weighting factor that can be used to tune preferences. The $Satisfies(p_i, I_s)$ function depends on the type of feedback the recommender supports. Let's describe a couple of examples: (1) In a content-based recommender system that uses *critiquing-based* feedback, this measure computes whether a critique[2] is satisfied by product p_i; (2) In a collaborative filtering recommender that uses *ratings-based* feedback, this measure directly returns the rating[3] value.

Next, we define different strategies for reaching a consensus in a group recommender system. The consensus recommended product or products —i.e., it also depends on the recommender— to a group is generated according to equation 2.

$$consensus(P, IM) = arg \max_{p_i \in P}(strategy(p_i, IM)) \tag{2}$$

where *strategy* refers to the name of the strategy.

3.2 Statistical dispersion

In this section we propose two strategies, called mean and purity, based on the measurement of dispersion used in statistics and probability theory.

The first consensus-based proposal of this paper is called *mean*. The mean satisfaction, $m(p_i, IM)$, of the group for a product, p_i, is defined as the sum of each member, u_j, individual satisfaction according to her preferences, IM^{u_j}.

$$m(p_i, IM) = \frac{1}{k} \sum_{j=1}^{k} \delta(p_i, IM^{u_j}) \tag{3}$$

The aim of Equation 3 is simply to offer a mean or average to derive a central tendency of the preference

space. On the other hand, the deviation measures the variability or diversity there is from the mean. The deviation measure is required for the second proposal of this paper: the purity.

$$d(p_i, IM) = \sqrt{\frac{1}{k-1} \sum_{j=1}^{k} \left(\delta(p_i, IM^{u_j}) - m(p_i, IM) \right)^2} \tag{4}$$

We use the sample standard deviation, $d(p_i, IM)$, as defined in Equation 4. It shows the deviation in the satisfaction of the users. A low measure indicates that the preferences tend to be close to the mean and a high value indicates that the preferences are spread out over a large range of values.

Purity is the second strategy of dispersion proposed in this paper, see Equation 5. Roughly, it measures the percentage of positive preferences among the whole set of preferences made by the group that are covered by the product.

$$purity(p_i, IM) = \frac{\sum_{j=1}^{k} \delta(p_i, IM^{u_j}) - d(p_i, IM)}{R} \tag{5}$$

A value $purity(p_i, IM) = 1$, means that all members of the group satisfy their preferences whereas a value $purity(p_i, IM) = 0$, denotes that none of the preferences of the group are satisfied. This strategy owes its name to a similar measure broadly used in rule-based learning [12]. It has commonly been used to find a rule that covers as many positive examples while covering as few negative examples as possible. The objective of any rule-based algorithm is to find a trade-off between these two conditions using the purity measure in different ways. We have made an adaptation of this measure to group recommender systems. In our case, we measure how many preferences are covered (i.e., satisfied) by a product while considering all the preferences of the group. Moreover, we have included the deviation in the equation to denote the dispersion of the satisfaction.

3.3 Individual content

This section describes three individual content strategies named completeness, logical sufficiency and group sufficiency. Individual content dispersion strategies aim at evaluating the individual satisfaction of the members.

The *completeness* measure has been previously used in negotiation scenarios. For example, in auctions [7] where a buyer and a provider want to reach agreement for the best offer. The negotiation has usually been considered between two individuals. In a group recommendation scenario the negotiation may involve more than

---

[2] A critique, I_s, indicates a directional feature-value preference. For example, *price* greater than $500.

[3] A rating, I_s, is a numerical score provided by the user that measures how much user u_j likes an item p_i. In this particular case, the preference provided also denotes the satisfaction of the user.

two individuals. Thus, Equation 6 has been modified accordingly to group recommenders. The objective of the completeness strategy is to favor high scores while penalizing big differences between members.

$$completeness(p_i, IM) = \frac{\sum_{j=1}^{k} w_j \cdot \sqrt{\delta(p_i, IM^{u_j})}}{\sum_{j=1}^{k} \sqrt{r_{u_j}}} \quad (6)$$

where w_j is a weighting factor that $\sum_{j=1}^{k} w_j = 1$ and $w_j \geq 0$. As defined in Equation 6, the completeness of a product, p_i, is computed in terms of members' satisfaction, $\delta(p_i, IM^{u_j})$. Moreover, the parameter w_j is a weighting factor used to tune the completeness – i.e., to focus more on one of the member's preferences. The sum of $\sqrt{r_{u_j}}$ factors is for normalizing the values in the range $[0,1]$.

The degree of *logical sufficiency* [11] (i.e, in short the *ls* measure) is a standard likelihood ratio statistic, which has been applied to measure rule quality of rule induction systems. This measure divides the proportion of positive examples that are covered by a rule by the proportion of negative examples. Equation 7 shows this measure adjusted for group recommendation. Given a product, p_i, the degree of logical sufficiency of p_i with respect to an individual user, u_j, is defined as the proportion of satisfaction of u_j by the proportion of u_j preferences that are not satisfied.

$$ls(p_i, IM^{u_j}) = \frac{\delta(p_i, IM^{u_j})}{r_{u_j} - \delta(p_i, IM^{u_j})} \quad (7)$$

Equation 7 measures the user confidence for a product considering her preferences. Note that a large *ls* means user u_j is widely satisfied by product p_i. Note that in the extreme *ls* approaches to infinity and, in this case, we restrict the value to $ls(p_i, IM^{u_j}) = \delta(p_i, IM^{u_j})$. On the other hand, if *ls* is much less than unity, then the product p_i is discouraging for user u_j. The degree of logical sufficiency of p_i for the whole group, ls_g, as shown in Equation 8, is defined as the sum of individual logical sufficiency divided by the total number of preferences in the group.

$$ls_g(p_i, IM) = \frac{\sum_{j=1}^{k} ls(p_i, IM^{u_j})}{R} \quad (8)$$

In inductive rule learning algorithms, the *ls* measure gives rise to the *lscontent* measure. Equation 9 shows the adaptation to group recommender systems. Essentially, *lscontent* estimates *ls* measure with a Laplace correction that penalizes products with low coverage.

$$lscontent(p_i, IM^{u_j}) = \frac{\delta(p_i, IM^{u_j}) + 1}{r_{u_j} - \delta(p_i, IM^{u_j}) + 1} \quad (9)$$

When we evaluate the degree of logical sufficiency of a product in relation to an individual member, there

is no differences between *lscontent* and *ls*. However, *lscontent* may help to reach a consensus when we apply the degree of logical sufficiency to describe a group sufficiency.

We also propose in this paper a *group sufficiency* —in short *gs*— measure based on *lscontent*. Instead of analyzing the logical sufficiency of a individual member in relation to her satisfied or unsatisfied preferences, *gs* measures the satisfaction of a member in relation to the satisfied preferences for the rest of the group and to the unsatisfied preferences for the individual in comparison with the whole group. The intuition behind this strategy comes from the fact that the satisfaction of an individual is likely to depend on that of other individual in the group (emotional contagion), as observed by [28].

$$gs(p_i, IM^{u_j}) = \frac{\frac{\delta(p_i, IM^{u_j}) + 1}{\sum_{s=1}^{k} \delta(p_i, IM^{u_s}) + 2}}{\frac{r_{u_j} - \delta(p_i, IM^{u_j}) + 1}{\sum_{s=1}^{k} r_{u_s} - \delta(p_i, IM^{u_s}) + 2}} \quad (10)$$

$$gs_g(p_i, IM) = \left(\frac{\sum_{j=1}^{k} gs(p_i, IM^{u_j})}{k} \right) \quad (11)$$

Equation 10 defines the group sufficiency for a given product p_i and a user u_j as the proportion of satisfied preferences for user u_j in front of the satisfied preferences of the whole group divided by those preferences that have not been satisfied of user u_j in front of those unsatisfied for the group. The Laplace correction in this case adds a factor of $\frac{1}{2}$ for penalizing products with low coverage. If a product covers no users, its Laplace will be $\frac{1}{2}$. Finally, the degree of group sufficiency of p_i is defined as the sum of individual group sufficiency divided by the number of users in the group, see Equation 11. This equation estimates as the best product the one that minimizes *gs*. Thus, instead of Equation 2, it uses Equation 12.

$$consensus(P, IM) = arg \min_{p_i \in P} (gs(p_i, IM)) \quad (12)$$

3.4 Collaborative Filtering

In this section we briefly describe the most well-known strategies used in collaborative filtering algorithms for recommendation generation. We argue that they may also be useful for consensus negotiation.

The *Least misery* strategy, as defined in Equation 13, chooses a product p_i based on the minimum satisfaction of the individual preferences. The rationale is that a group is as satisfied as its least satisfied member. POLYLENS [35] uses this strategy for a group of people that is going to watch a movie together.

$$least\_misery(p_i, IM) = arg \min_{u_j \in U} (\delta(p_i, IM^{u_j})) \quad (13)$$

Most pleasure strategy selects the maximum satisfaction of the individual preferences, see Equation 14. It considers that at least one member will be maximally satisfied.

$$most\_pleasure(p_i, IM) = arg \max_{u_j \in U} (\delta(p_i, IM^{u_j})) \quad (14)$$

Previous strategies assume the consensus based on the satisfaction of one individual: the least or the most satisfied. There is no warranty that the recommendations will suit the whole group.

Multiplicative strategy multiplies the satisfaction of the individual users, see Equation 15. With this strategy it might happen that a member with unique tastes always lose out because their opinion happens to be a minority preference.

$$multiplicative(p_i, IM) = \prod_{j=1}^{k} (\delta(p_i, IM^{u_j})) \quad (15)$$

Finally, we describe the *Borda count* strategy. The *Borda count* is a voting system divided into two steps. First of all, each individual user ranks the products in order of her individual preferences. Thus, the product p_i for individual u_j will obtain a rank value that is denoted as $rank_{u_j}^{p_i}$. The rank values are computed as follows: (1) Sort out the products according to the satisfaction of the user u_j; (2) Assign to the first product one point, second product receives two points, third product three points and so on. In case there is two or more products with the same satisfaction, the points are averaged and distributed among the products. Second step is devoted to sum rank values obtained for the members of the group, see Equation 16.

$$borda\_count(p_i, IM) = \sum_{j=1}^{k} (rank_{u_j}^{p_i}) \quad (16)$$

The borda count determines the winner product (i.e., the consensus product) by giving each candidate product a certain number of points corresponding to the position it is ranked considering the satisfaction of each user. This strategy has been previously analyzed as a recommendation generation strategy in [2,27].

4 An illustrative example

To help understand the strategies, we present a simple example to illustrate the consensus-based recommendations suggested by each strategy. We consider the consensus process for a group of four individuals (i.e., $U = \{Paul, Anne, Mary, John\}$, then $k = 4$) and a product base of ten products, $P = \{A, B, ..., J\}$. For

the sake of simplicity, we consider that all users contains a individual user model with ten preferences (i.e., $R = 10 + 10 + 10 + 10 = 40$) and that the satisfaction, δ, of these preferences is shown in Table 1. For example, Table 1 shows that Paul and Mary's ten preferences are satisfied with product A, while the same product only satisfies four of Anne's preferences with seven of John's preferences being satisfied. We have also included the sum of preferences satisfied by each product. Looking at product A and F, both satisfy 31 of the group preferences. However, each one of them shows a different satisfaction of the users' preferences.

Table 1 Example of the satisfaction for a group of four individuals

	A	B	C	D	E	F	G	H	I	J
Paul	10	4	3	6	10	9	6	8	10	8
Anne	4	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	7	5	6	7	6
John	7	6	9	8	6	6	10	9	9	9
sum	31	24	22	30	32	31	27	32	29	31

The ls_g, gs_g and *borda_count* strategies are computed in two phases. First one is devoted to compute a partial result and the second one uses this partial result to obtain a final value. Table 2 shows the results in the first phase for *ls*, *gs* and the rank computation of the *borda_count*, respectively.

According to the example described in Table 1, we compute each one of the consensus-based strategies to select a set of products for consensus recommendation[4]. In this example, the set will contain six products. Table 3 shows the results of applying all the consensus strategies. A row indicates for a particular consensus strategy, which is the value obtained when applying the strategy to every one of the products considering all the individual user models. For example, the *mean* row indicates for product ($p_i = A$) (i.e., see column A) that $mean(A, IM) = \frac{10+4+10+7}{4} = 7,75$.

Last column in Table 3 depicts the consensus set of products recommended by each strategy. One or more products in parenthesis means that there is a draw between them. Table 3 shows that nearly all strategies select the set of six products in a different order. It is interesting to note that most of strategies agree that H is the best product. However, the second option is not so clear. For example, *purity*, *completeness*, *most_pleasure* and *multiplicative* choose product E and, *least misery* selects product J.

---

[4] As previously mentioned, depending on the recommender, the set of recommendations may be one or more products.

Table 2 Results for each individual member for the *ls*, *gs* and the ranks for the *borda_count*, respectively.

		A	B	C	D	E	F	G	H	I	J
ls	Paul	10	0,67	0,43	1,5	10	9	1,5	4	10	4
	Anne	0,67	9	4	9	2,33	9	1,5	9	0,43	4
	Mary	10	1	0,25	2,33	9	2,33	1	1,5	2,33	1,5
	John	2,33	1,5	9	4	1,5	1,5	10	9	9	9
gs	Paul	3,67	0,49	0,42	0,53	3,24	1,67	0,72	0,88	4,61	1,00
	Anne	0,24	3,46	2,50	1,88	0,59	1,67	0,72	1,47	0,21	1,00
	Mary	3,67	0,69	0,28	0,75	1,47	0,67	0,52	0,41	0,84	0,47
	John	0,67	0,97	4,17	1,13	0,41	0,47	5,69	1,47	2,10	1,67
ranks	Paul	9,0	2,0	1,0	4,0	9,0	7,0	3,0	5,5	9,0	5,5
	Anne	2,0	8,5	5,5	8,5	4,0	8,5	3,0	8,5	1,0	5,5
	Mary	10,0	2,5	1,0	7,5	9,0	7,5	2,5	4,5	6,0	4,5
	John	4,0	2,0	7,5	5,0	2,0	2,0	10,0	7,5	7,5	7,5

Table 3 Consensus-based strategies.

	A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**Consensus**
mean	7,750	6	5,5	7,5	8	7,75	6,75	8	7,25	7,75	H, E, J, F, A, D
purity	0,759	0,588	0,531	0,743	0,790	0,767	0,663	0,792	0,708	0,768	H, E, J, F, A, D
completeness	0,867	0,766	0,710	0,864	0,890	0,877	0,814	0,892	0,833	0,878	H, E, J, F, A, D
ls_g	0,575	0,304	0,342	0,421	0,571	0,546	0,350	0,588	0,544	0,463	H, A, E, F, I, J
gs_g	2,060	1,404	1,840	1,069	1,426	1,117	1,914	1,059	1,940	1,033	J, H, D, F, B, E
least_misery	4	4	2	6	6	6	5	6	3	6	(H, J, E, F, D), G
most_pleasure	10	9	9	9	10	9	10	9	10	9	(A, E, I, G), (H, F)
multiplicative	2800	1080	432	3024	3780	3402	1800	3888	1890	3456	H, E, J, F, D, A
borda_count	25	15	15	25	24	25	18,5	26	23,5	23	H, (F, D, A), E, I

Additionally, Table 3 also shows that *purity* and *completeness* select identical set of products. Coincidences among strategies may happen because this example contains a reduced set of ten products, the size of the group is small and the users are quite similar in their preferences. The next section is devoted to the in-depth analysis of the strategies for different types of user groups, ranging from very similar groups to more diverse groups with competing preferences. Moreover, it also analyzes the strategies when varying the group size.

5 Evaluation

Ideally we would like to evaluate our consensus negotiation strategies through a large-scale live-user study, however, we wish to compare an array of strategies across a range of group sizes made up of members with various preference similarities. For this reason a live-user study was unfeasible. Artificially modelling user preference information and behaviour is difficult and sometimes unreliable. In this evaluation we have endeavoured to utilise real user data in an artificial setting by combining live-user preference information and simulating the consensus negotiation of the groups.

In order to generate groups and test our consensus negotiation strategies we need user profiles made up of preference information. In this instance, we opt for a critiquing-based recommender style user profile of critique preferences [31], see Section 5.1. Critiques are directional preferences applied to a particular feature of a product, e.g. *Price < $100* or *Rating > 4 Stars*. In this evaluation, we will use the critiquing profiles as input to the consensus negotiation strategies.

5.1 Description of the recommender

In this work, although we will describe an artificial simulation evaluation, our profiles are generated using real user data in a fashion which reflects a live user group recommender system called CATS. The Collaborative Advisory Travel System (CATS) [31], is a prototype recommender system that supports consensus decision-making for a group of users intending to book a ski-holiday together. Very briefly, in CATS, holiday candidates are represented as product cases, each describing various features about the resort and ski-runs available. Sample product features include: *package price, number of ski runs/difficulty, location, accommodation type/rating* and *experience level of the skier*. CATS uses

a common interface to share preferences among group members (see Figure 1). Importantly, a number of novel and interesting mechanisms have been put in place to maximize the amount of preference information captured from, as well as communicated to, group members (see [30–32]). This is so that the recommender can uncover useful information about the combined preferences of the group and make more appropriate recommendations. Group recommendation cycles start with individual group members expressing their preferences over holiday products, CATS generates individual and group recommendations by aggregating the evolving preference profiles. The recommendations are presented to the group members to allow for feedback. In this group recommender users provide feedback on holiday products using critique-based feedback [6]. The CATS system was developed to be device unspecific and prototypes were developed for multiple PCs connected over a network [31] and a table-top collaborative display [32] (see Figure 1).

5.2 Data and Users

For our evaluation we use a product-base of 151 European ski packages as our product cases. Each product is made up of 42 different features related to the ski resort (25 features such as *country, transfer time, lift system, etc.*) and the type of accommodation (17 features such as *rating, price, ski room facilities, etc.*). As our trialists we enlisted the help of 34 postgraduate students with a range of skiing experience. Of the participants, 7 users had skied regularly before and so were very aware of their skiing preferences, while the other 27 users were novices or first-timers with a more limited idea of their preferences (see also [31]).

5.3 Methodology

The style of this evaluation is different from others in that our test users do not participate in a live evaluation of different versions of a recommendation system as members of a set number of well-defined groups. Our goal is to evaluate group consensus negotiation recommendation across a large number of different groups with very different characteristics. Ordinarily this would mean enlisting the help of large numbers of users, which was judged to be prohibitively expensive. Therefore, instead we chose to spend significant effort eliciting the ski preferences of our test subjects and used these preferences as the basis for an off-line evaluation by combining the users into large numbers of different test groups. These test groups (and their members' preferences) are used to evaluate the recommendations made by the different group consensus negotiation strategies.

5.4 Critique-Based Profiling

In our group recommender scenario users provide feedback on holiday products (also known as *cases* in conversational case-based recommenders) using critique-based feedback [5]. When presented with a recommendation the users apply contextual critiques to product features in line with their preferences. For instance, they might seek a holiday that is less/more expensive, a higher/lower standard of accommodation, or one which provides access to more/less advanced ski runs.

In order to produce these critique profiles user preferences were recorded as each test subject browsed a collection of sample ski products [31]. No recommendation techniques were used during this phase as the objective was simply to allow the user to review the available holiday options and select a single preferred product (their *Final Product*) in their own time. Trial subjects were also asked to complete a Web form indicating which of the *Final Product* features they felt positively or negatively about in order to get a clearer picture of their preferences with respect to this product. Users were also asked to design their own "*Perfect Product*" by completing a Web form to fill out their ideal set of ski holiday features. Users were instructed to make reasonable choices during this stage; it is unreasonable to expect a week in a 5-star hotel for $100, for example. Of course there are no guarantees that the resulting "product" will exist in the product space — in fact it is highly unlikely — but it provides us with a clear picture of each user's true preferences, broadly unconstrained by the reality of what is available. At the end of this phase, the 34 trial subjects had chosen a total of 26 unique *Final Products* after reviewing an average of 26 products each (i.e., the typical user chose their preferred product after viewing approximately 17% of the available products). On average each user annotated 11 features of their *Final Product* as positive and 3 features as negative and when they produced their *Perfect Product* they selected an average of 14 features.

Each of the preference profiles for the 34 trial subjects was then converted into a critique-based profile that could be used by our recommender system. To do this we inferred a set of critiques by comparing each user's *Final Product* features (positive and negative) to their corresponding *Perfect Product* features. For example, if a user indicated a positive preference for (*price = $1000*) and their corresponding *Perfect Case* feature for (*price = $800*) the inferred critique would be (*price < $1000*). Also for a negative preference, for example
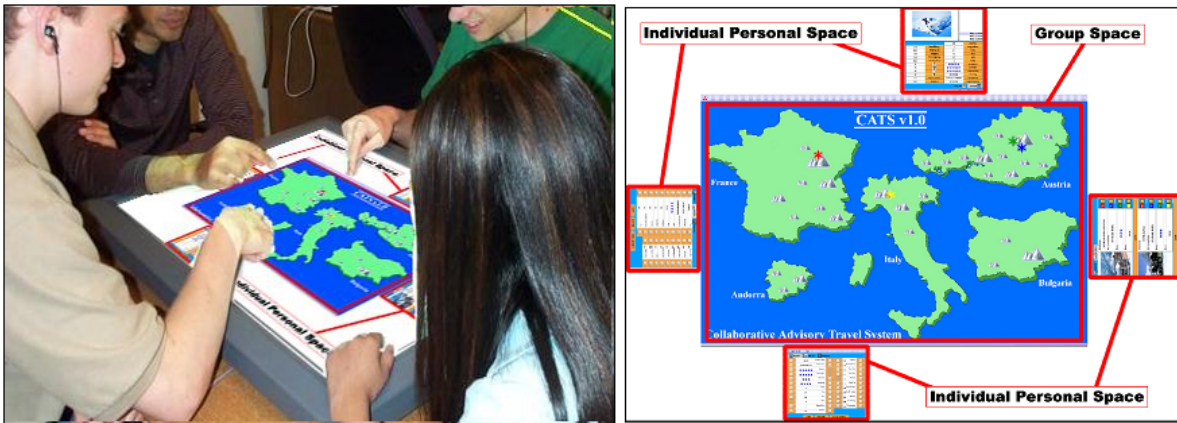
Fig. 1 Illustrating the CATS interaction with a table-top display.

($Ensuite = No$), an ($Ensuite = Yes$) critique would be generated. The result was a profile for each user made up of a set of feature-value critiques. On average, each profile contained just over 7 nominal critiques and 3-4 ordinal critiques; specifically, 6.029 *Equal to Nominal* critiques, 1.098 *Not Equal to Nominal* critiques, 1.059 *Less than Ordinal* critiques, 2.471 *Greater than Ordinal* critiques and 2.441 *Equal to Ordinal* critiques.

5.5 Group Generation

From our set of 34 profiles we can generate combinatorially many groups made up of users with varying degrees of similarity. Firstly we generated groups of various sizes, and in this case our groups are made up of 3, 4, 6 and 8 individual members. We wish to see if the consensus negotiation strategies performance change with varying group size. We generated several thousand groups from our profiles and calculated the group members average similarity. To compute inter-user similarity we compared users critique satisfaction with their *Perfect Products*. We then split these groups to form 3 sets of 100 groups. Each set of groups was made up of members with certain similarity characteristics. For example, the *Similar* group-set contained groups of users with a high average pairwise similarity in the range of 0.53 to 0.82. The *Mixed* group-set contained groups of users with a mid-range pairwise similarities in the region of 0.4. Finally, the *Diverse* group-set contains groups of users with average pairwise similarities from 0.04 to 0.28. Intuitively we would expect that more diverse groups will represent more challenging consensus targets as their individual members tend to have conflicting preferences.

In order to compare our consensus negotiation strategies we need a baseline. There has been little or no other published strategies for consensus negotiations for these types of group recommender systems, so a challenging baseline is difficult to locate. In this evaluation we compare our strategies against each other and the Random baseline – a product returned at random from the product space.

5.6 Consensus Negotiation

For each test group we generate consensus negotiation recommendations across all the strategies described in Section 3 and our baseline. Each simulation recommends a final list of recommendations to the group. We will evaluate consensus negotiation through the recommendation of a single top recommendation and a final recommendation list of 5 products. Typical recommender systems differ in how they display final recommendation lists with some systems preferring a single final product and others offering some choice by narrowing the options to a number of possibilities.

Figure 2 shows the results for the top 1 consensus negotiation recommendation for each strategy across all of the group sizes broken down by group preference similarity. The bars represent the average similarity of the recommendation product to the *Perfect Products* of the group members. The graphs show similar trends for each of the group sizes allowing us to conclude that consensus negotiation is stable across different group sizes. As expected the Similar groups were easier to satisfy than the Mixed or Diverse groups. The graph shows that the Completeness, Multiplicative, Borda and Mean strategies perform best across all group sizes. LSg, Purity and Least Misery all perform admirably. Overall the best of the strategies produce a single recommendation which is between 60% and 80% similar to each members *Perfect Products*. All strategies outperform the Random baseline.
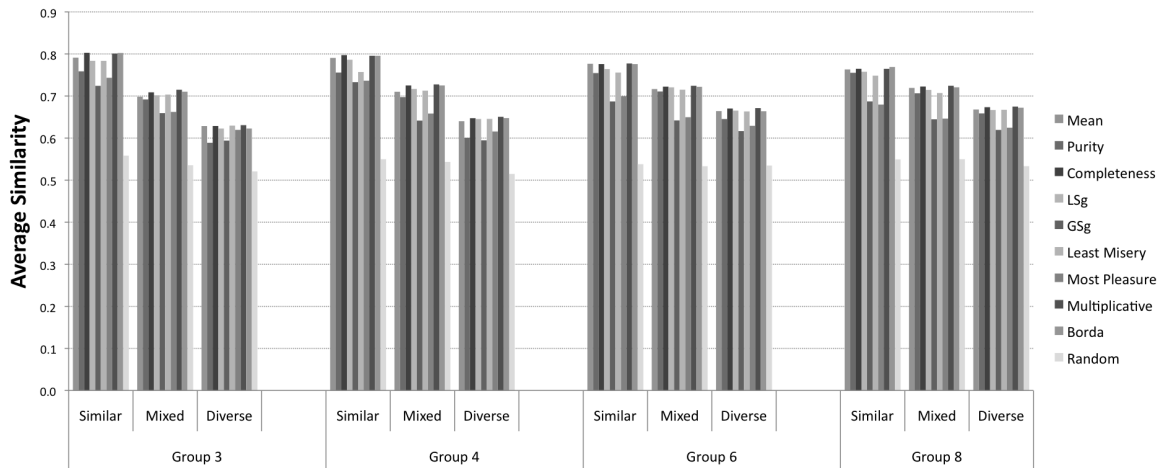
Fig. 2 Average Perfect Product Similarity for Top 1 Recommendation.
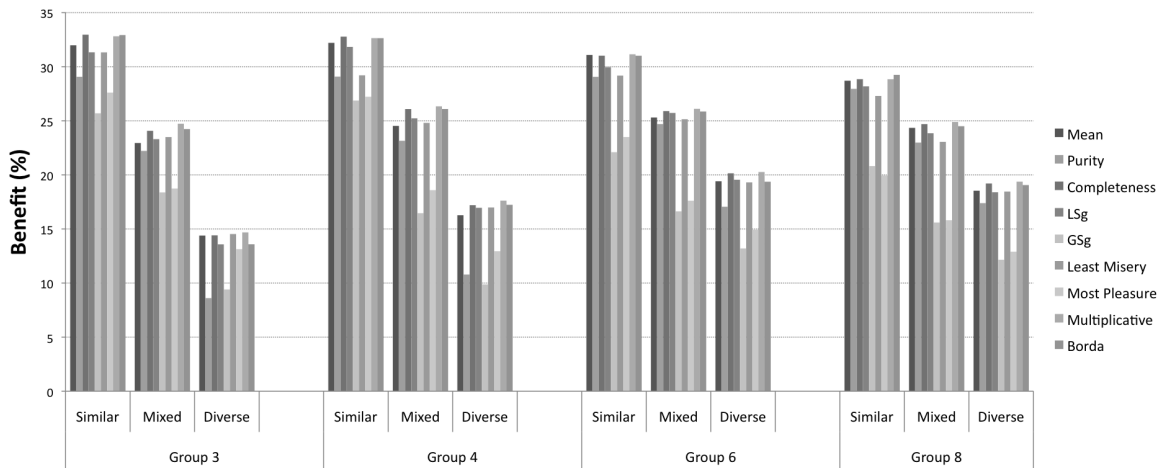

Fig. 3 Percentage Benefit for Each Strategy for Top 1 Recommendation.

Figure 3 shows the percentage benefit of each strategy over the Random baseline. Again, as expected, the Similar groups receive the greatest share of the benefit. Interestingly however, as the group size gets larger the more Diverse groups receive a greater benefit, with Multiplicative and Completeness offering the most for the most diverse 8 member groups.

In Figure 4 we examine the results when the top 5 consensus negotiation recommendations are offered to the group. The bars in these graphs represent the average similarity of the top 5 recommendations to the *Perfect Products* of the group members. The results for the top 5 recommendations are broadly similar to those produced in the single recommendation analysis with Multiplicative, Borda Count and Completeness performing best, closely followed by Purity, Least Misery and LSg. Again, the strategies are stable across various group sizes and various inter-member similarities and all outperform the Random baseline. Once again,

the result trends are similar when we look at the percentage benefit of the strategies over the Random baseline (Figure 5). So not only are the results stable across differing inter-member similarity of various size groups for a single top 1 consensus recommendation, but they are also stable and offer similar benefits when a list of top 5 consensus products are recommended.

5.7 Statistical analysis

We also statistically analyze the benefits of using each one of the strategies. First of all, we compute the *mean rank* (r) of each strategy considering all the experiments (three different types of groups with four group sizes). The rankings are obtained estimating each particular ranking r_i^j for each experiment i and each strategy j, and computing the mean ranking R for each strategy as $R_j = \frac{1}{N} \sum_i r_i^j$, where N is the total number of experiments. Compared with mean performance val-
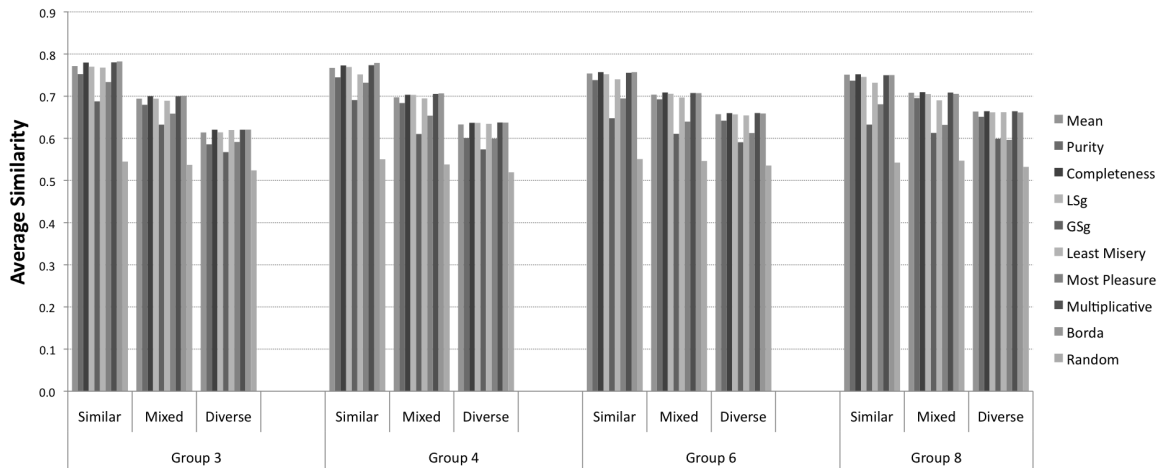
Fig. 4 Average Perfect Product Similarity for Top 5 Recommendations.
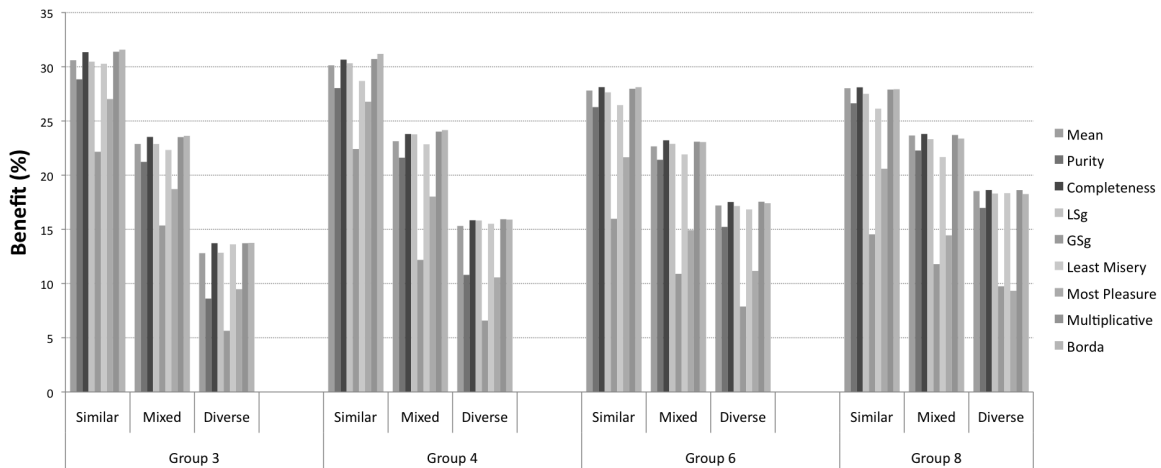

Fig. 5 Percentage Benefit for Each Strategy for Top 5 Recommendation.

ues, the mean rank reduces the susceptibility to outliers which, for instance, allows a classifier's excellent performance in one experiment to compensate for its overall bad performance [10]. Secondly, we apply the Friedman and Nemenyi tests to analyze whether the difference between algorithms is statistically significant [13, 14].

The **Friedman test**, recommended by Demšar [10], is effective for comparing multiple strategies or algorithms across multiple data sets (in our case, across multiple experiments). It compares the mean ranks of strategies to decide whether to reject the null hypothesis, which states that all the methods are equivalent and therefore their ranks should be equal. The Friedman statistic value is computed as follows:

$$X_F^2 = \frac{12N}{k(k+1)} \left[\sum_j R_j^2 - \frac{k(k+1)^2}{4} \right] \qquad (17)$$

where k is the number of strategies to compare.

Since this value is undesirably conservative, Iman and Davenport [18] proposed a corrected statistic:

$$F_F = \frac{(N-1)X_F^2}{N(k-1) - X_F^2} \qquad (18)$$

When we apply the Friedman test in our experiments with ten strategies and twelve different experiments, F_F is distributed according to the F distribution with $(10 - 1) = 9$ and $(10 - 1) \cdot (12 - 1) = 99$ degrees of freedom. The critical value of F(9,99) = 2,59 at the 0,01 critical level. For our experiments on the Ski dataset we obtained the values of $X_F = 97, 65$ and $F_F = 103, 78$ and, $X_F = 95, 57$ and $F_F = 84, 60$ for the Top 1 and Top 5 experiments, respectively. As the values are higher than 2,59 we can reject the null hypothesis in both cases.

Once we have checked for the non-randomness of the results, we can perform a post hoc test to check if one of the techniques can be singled out. For this purpose, we use the Nemenyi test —two techniques are significantly
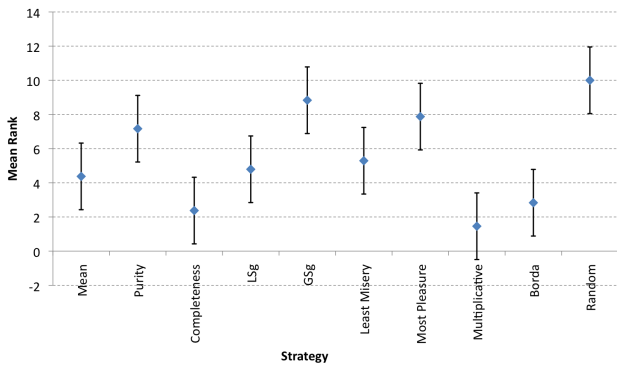
Fig. 6 Application of the Nemenyi test to the consensus strategies when they return the top product (top 1).

Fig. 7 Application of the Nemenyi test to the consensus strategies when they return the top 5 product (top 5).

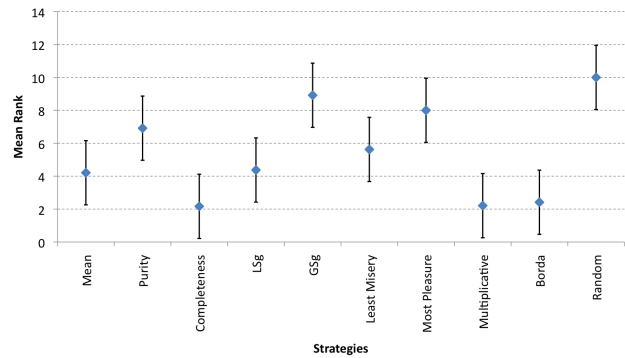different if the corresponding average ranks differ by at least the critical difference value:

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \qquad (19)$$

where q_α is based on the Studentized range statistic divided by $\sqrt{2}$ and k is the number of algorithms to compare. In our case, when comparing ten strategies with a critical value $\alpha = 0.05$, $q_{0.05} = 3.164$ for a two-tailed Nemenyi test. Substituting, we obtain a critical difference value $CD = 3.91$. Thus, for any two pairs of algorithms whose rank difference is higher than 3.91, we can infer —with a confidence of 95%— that there exists a significant difference between them.

For the top 1 product recommendation, the results of the Nemenyi test are illustrated in Figure 6. In the figure, diamonds represent the mean ranks of each strategy. Vertical lines across diamonds indicate the 'critical difference'. The performance of two strategies is significantly different if their corresponding mean ranks differ by at least the critical difference. For instance, Figure 6 reveals that Completeness is significantly better than Purity. However, we cannot say the same with regard to Mean or LS_g, though. Note that the best mean rank corresponds to Multiplicative strategy, closely followed by Completeness and Borda. Random strategy obtains the worst mean rank, as expected. Multiplicative, Completeness and Borda performance are significantly better than Purity, GSg, Most Pleasure and Random strategies.

A similar behavior occurs with the top 5 analysis of Figure 7. However, in this case, the best mean rank corresponds to the Completeness strategy, closely followed by the Multiplicative and Borda.

In summary, our statistical significance analysis has shown that Multiplicative, Completeness and the Borda strategies work particularly well across different group sizes and different types of users, as the three strategies present the lowest mean rank. Moreover, the results also show that these strategies are significantly better than Purity, GS_g, Most Pleasure and random strategies. On the other hand, it is interesting to note that Average, LS_g and Least misery present a greater mean rank than Completeness, Multiplicative and Borda although they are not significant differences among them. We can conclude that these strategies are well suited for consensus as they are stable across different combinations of users.

6 Conclusions

In this paper, we have introduced the little considered problem of consensus negotiation in group recommender systems. These systems differ from normal personalization recommenders as they aim to recommend personalized products to a group of users. These groups can be of various sizes and can be composed of members who may have conflicting product preferences. All of these possible variances make for an interesting research challenge. Here we have introduced nine consensus negotiation strategies to help the personalization process at the end of a session in a group scenario. The consensus negotiation strategies introduced are based on statistical, content and collaborative ideas. We carried out a large artificial simulation using real user preference information based on a ski holiday group recommender system employing critique-based feedback. The results are positive, with some significance, and the strategy behaviors are consistent across various group size and inter-member similarity. To date the issue of consensus negotiation has remained an acknowledged but unresolved challenge in this area of personalization. This work represents the first step in tackling this area of research.

As a matter of future work we would like to try and perform a deeper analysis of the behaviors of group consensus negotiation strategies. Although a full scale live-user study may still be premature, we wish to con-

tinue to develop our off line simulator and our ability to capture real user preference profiles for the purposes of introducing and testing new group recommendation ideas.

References

1. Ardissono, L., Goy, A., Petrone, G., Segnan, M., Torassol, P.: In-trigue: Personalized recommendation of tourist attractions for desktop and handset devices. Applied Artifcial Intelligence **17**(8-9), 687–714 (2003)
2. Baltrunas, L., Makcinskas, T., Ricci, F.: Group recommendations with rank aggregation and collaborative filtering. In: Proceedings of the fourth ACM conference on Recommender systems, pp. 119–126. ACM (2010)
3. Berkovsky, S.: Ubiquitous user modeling in recommender systems. In: User Modeling 2005, *Lecture Notes in Computer Science*, vol. 3538, pp. 496–498. Springer Berlin / Heidelberg (2005)
4. Berkovsky, S., Freyne, J.: Group-based recipe recommendations: analysis of data aggregation strategies. In: Proceedings of the fourth ACM conference on Recommender systems, pp. 111–118. ACM (2010)
5. Burke, R., Hammond, K., Young, B.: The FindMe Approach to Assisted Browsing. Journal of IEEE Expert **12**(4), 32–40 (1997)
6. Burke, R., Hammond, K., Young, B.: The FindMe Approach to Assisted Browsing. Journal of IEEE Expert **12**(4), 32–40 (1997)
7. Cerquides, J., López-Sánchez, M., Reyes-Moro, A., Rodríguez-Aguilar, J.A.: Enabling assisted strategic negotiations in actual-world procurement scenarios. Electronic Commerce Research **7**, 189–220 (2007)
8. Crossen, A., Budzik, J., Hammond, K.J.: Flytrap: Intelligent group music recommendation. In: Y. Gil, D.B. Leake (eds.) Proceedings of The International Conference on Intelligent User Interfaces (IUI-2002), pp. 184–185. ACM, New York, USA (2002)
9. De Silva, H., Moessener, K., Carrez, F.: Group knowledge management for context-aware group applications and services. In: Personal, Indoor and Mobile Radio Communications, 2009 IEEE 20th International Symposium on, pp. 2851 –2855 (2009)
10. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal Machine Learning Research **7**, 1–30 (2006)
11. Duda, R., Gaschnig, J., Hart, P.: Model design in the prospector consultant system for mineral exploration. In: D. Mitchie (ed.) Expert systems in the micro-electronic age. Edinburgh, UK. Edinburgh University Press (1979)
12. Ferrer-Troyano, F., Aguilar-Ruiz, J.S., C., R.J.: Incremental rule learning based on example nearness from numerical data streams. In: Proceedings of the 2005 ACM symposium on Applied computing, SAC '05, pp. 568–572. ACM (2005)
13. Friedman, M.: The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the American Statistical Association **32**(200), 675–701 (1937)
14. Friedman, M.: A comparison of alternative tests of significance for the problem of m rankings. The Annals of Mathematical Statistics **11**(1), 86–92 (1940)
15. Goren-Bar, D., Glinansky, O.: Family stereotyping - a model to filter tv programs for multiple viewers. In: Proceedings of the Workshop on Personalization in Future TV at the Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH-2002). Springer Verlag, Malaga, Spain (2002)
16. Gratz, P., Andronache, A., Rothkugel, S.: Ad hoc collaborative filtering for mobile networks. In: Proceedings of the 2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (sutc 2008), pp. 355–360. IEEE Computer Society (2008)
17. Haensel, R., Coutand, O., Kernchen, R., Nanni, R., Moesser, K., Droegehorn, D.: Time gems: Group experience in pervasive computing environments. In: Proc of the 15th IST Mobile and Communications Summit (2006)
18. Iman, R., Davenport, J.: Approximations of the critical region of the friedman statistic. Communications in statistics pp. 571–595 (1980)
19. 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, pp. 48–54. Gallipoli, Italy (2004)
20. Jameson, A., Baldes, S., Kleinbauer, T.: Enhancing mutual awareness in group recommender systems. In: B. Mobasher, S. Anand (eds.) Proceedings of the IJCAI 2003 Workshop on Intelligent Techniques for Web Personalization. AAAI, Menlo Park, CA (2003)
21. Jameson, A., Smyth, B.: Recommendation to groups. In: P. Brusilovsky, A. Kobsa, W. Nejdl (eds.) The adaptive web, pp. 596–627. Springer-Verlag, Berlin, Heidelberg (2007)
22. Jung, K.Y.: User preference through learning user profile for ubiquitous recommendation systems. In: Knowledge-Based Intelligent Information and Engineering Systems, *Lecture Notes in Computer Science*, vol. 4251, pp. 163–170. Springer Berlin / Heidelberg (2006)
23. Kim, H.K., Kim, J.K., Ryu, Y.U.: Personalized recommendation over a customer network for ubiquitous shopping. IEEE Trans. Serv. Comput. **2**, 140–151 (2009)
24. Kourouthanassis, P., Spinellis, D., Roussos, G., Giaglis, G.M.: Intelligent cokes and diapers: Mygrocer ubiquitous computing environment. In: In: Proceedings of The First International Conference on Mobile Business (M-business '02 (2002)
25. Lewis, N.: From customization to ubiquitous personalization: digital identity and ambient network intelligence. interactions **11**, 48–50 (2004)
26. Lieberman, H., Dyke, N.V., Vivacqua, A.: Let's browse: A collaborative web browsing agent. In: Proceedings of the International Conference on Intelligent User Interfaces (IUI-1999), p. 65. New York (1999)
27. Masthoff, J.: Group modeling: Selecting a sequence of television items to suit a group of viewers. User Model. User-Adapt. Interact. **14**(1), 37–85 (2004)
28. Masthoff, J., Gatt, A.: In pursuit of satisfaction and the prevention of embarrassment: affective state in group recommender systems. User Modeling and User-Adapted Interaction **16**, 281–319 (2006)
29. McCarthy, J., Anagnost, T.: Musicfx: An arbiter of group preferences for computer aupported collaborative workouts. In: Proc. of Conference on Computer Supported Cooperative Work, pp. 363–372 (1998)

30. Mccarthy, K., Mcginty, L., Smyth, B.: Case-based group recommendation: Compromising for success. In: Proceedings of the 7th international conference on Case-Based Reasoning: Case-Based Reasoning Research and Development, pp. 299–313. Springer-Verlag, Berlin, Heidelberg (2007)

31. McCarthy, K., McGinty, L., Smyth, B., Salamó, M.: The Needs of the Many: A Case-Based Group Recommender System. In: T.R. Roth-Berghofer, M.H. Goker, H.A. Guvenir (eds.) Proceedings of the European Conference on Case Based Reasoning (ECCBR 2006), pp. 196–210. Springer Verlag (2006). Turkey

32. McCarthy, K., Salamó, M., Coyle, L., McGinty, L., Nixon, B.S..P.: CATS: A Synchronous Approach to Collaborative Group Recommendation. In: Proceedings of the FLAIRS 2006 Conference, pp. 1–16. Springer Verlag (2006). Florida, USA

33. McGinty, L., Smyth, B.: Tweaking Critiquing. In: Proceedings of the Workshop on Personalization and Web Techniques at the International Joint Conference on Artificial Intelligence. Morgan-Kaufmann (2003)

34. Miller, B.N., Albert, I., Lam, S.K., Konstan, J.A., Riedl, J.: Movielens unplugged: experiences with an occasionally connected recommender system. In: Proceedings of the 8th international conference on Intelligent user interfaces, pp. 263–266. ACM, New York, NY, USA (2003)

35. O'Connor, M., Cosley, D., Konstan, J., Riedl, J.: PolyLens: A Recommender System for Groups of Users. In: Proc. of European Conference on Computer-Supported Cooperative Work, pp. 199–218 (2001)

36. Park, M.H., Park, H.S., Cho, S.B.: Restaurant recommendation for group of people in mobile environments using probabilistic multi-criteria decision making. In: Computer-Human Interaction, *Lecture Notes in Computer Science*, vol. 5068, pp. 114–122. Springer Berlin / Heidelberg (2008)

37. Pizzutilo, S., Carolis, B.D., Cozzolongo, G., Ambruoso, F.: Group modeling in a public space: Methods, techniques and experiences. In: Proceedings of WSEAS AIC 05. ACM, Malta (2005)

38. von Reischach, F., Guinard, D., Michahelles, F., Fleisch, E.: A mobile product recommendation system interacting with tagged products. In: Proceedings of the 2009 IEEE International Conference on Pervasive Computing and Communications, pp. 1–6. IEEE Computer Society (2009)

39. von Reischach, F., Michahelles, F., Schmidt, A.: The design space of ubiquitous product recommendation systems. In: Proceedings of the 8th International Conference on Mobile and Ubiquitous Multimedia, MUM '09, pp. 2:1–2:10. ACM, New York, NY, USA (2009)

40. Resatsch, F., Karpischek, S., Sandner, U., Hamacher, S.: Mobile sales assistant: Nfc for retailers. In: Proceedings of the 9th international conference on Human computer interaction with mobile devices and services, MobileHCI '07, pp. 313–316. ACM (2007)

41. Seitz, C., Berger, M., Bauer, B.: Towards a general approach to mobile profile based distributed grouping. Personal Ubiquitous Comput. **9**, 90–99 (2005)

42. Senot, C., Kostadinov, D., Bouzid, M., Picault, J., Aghasaryan, A., Bernier, C.: Analysis of strategies for building group profiles. In: User Modeling, Adaptation, and Personalization, Lecture Notes in Computer Science, pp. 40–51. Springer Berlin / Heidelberg (2010)

43. van Setten, M., Pokraev, S., Koolwaaij, J.: Context-aware recommendations in the mobile tourist application compass. In: P. De Bra, W. Nejdl (eds.) Adaptive Hyper-media and Adaptive Web-Based Systems, *Lecture Notes in Computer Science*, vol. 3137, pp. 515–548. Springer Berlin / Heidelberg (2004)

44. Smyth, B., Balfe, E., Freyne, J., Briggs, P., Coyle, M., Boydell, O.: Exploiting query repetition and regularity in an adaptive community-based web search engine. User Modeling User-Adapted Interaction **14**(5), 383–423 (2004)

45. Smyth, B., McGinty, L.: An Analysis of Feedback Strategies in Conversational Recommender Systems. In: P. Cunningham (ed.) Proceedings of the 14th National Conference on Artificial Intelligence and Cognitive Science (2003). Dublin, Ireland.

46. Wang, C.Y., Wu, Y.H., Chou, S.C.: Toward a ubiquitous personalized daily-life activity recommendation service with contextual information: a services science perspective. Information Systems and E-Business Management **8**, 13–32 (2010)

47. Wood, K.R., Richardson, T., Bennett, F., Harter, A., Hopper, A.: Global teleporting with java: Towards ubiquitous personalised computing. IEEE Computer **30**(2), 53–59 (1997)

48. Yu, Z., Zhou, X., Hao, Y., Gu, J.: Tv program recommendation for multiple viewers based on user profile merging. User Modeling and User-Adapted Interaction **16**, 63–82 (2006)

49. Zenebe, A., Ozok, A., Norcio, A.F.: Personalized recommender systems in e-commerce and m-commerce: A comparative study. In: Proc. of the 11th International Conference on Human-Computer Interaction (2005)