Interactive Multi-Party Critiquing for Group Recommendation

Francesca Guzzi
Free University of Bozen-Bolzano
Piazza Domenicani 3
Bozen-Bolzano, Italy
francesca.guzzi@gmail.com

Francesco Ricci
Free University of Bozen-Bolzano
Piazza Domenicani 3
Bozen-Bolzano, Italy
fricci@unibz.it

Robin Burke
DePaul University
Center for Web Intelligence
School of Computing
Chicago, IL, USA
rburke@cs.depaul.edu

ABSTRACT

Group recommender systems (RS) are used to support groups in making common decisions when considering a set of alternatives. Current approaches generate group recommendations based on the users' individual preferences models. We believe that members of a group can reach an agreement more effectively by exchanging proposals suggested by a conventional RS. We propose to use a critiquing RS that has been shown to be effective in single-user recommendation. In the group recommendation context, critiquing allows each user to get new recommendations similar to the proposals made by the other group members and to communicate the rationale behind their own counterproposals. We describe a mobile application implementing the proposed approach and its evaluation in a live user experiment.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*information filtering*

General Terms

Experimentation, Human Factors

Keywords

Group recommender system, decision making, critiquing

1. INTRODUCTION

Recommender systems (RSs) support users in finding items that best suit their preferences. In order to retrieve products, RSs estimate whether a user would or would not like a certain item. Group recommender systems address decision problems that involve more than one user, such as choosing a movie, a concert, or a restaurant [2].

The most common approaches to group recommendation start with profiles that represent individual user preferences. Such systems either (a) combine the user profiles and then generate recommendations for the aggregated profile, or (b) generate individual recommendations separately and aggregate the results [2]. However, research in social sciences and psychology has shown that there is no ideal method to aggregate individual preferences to generate common recommendations [3]. Moreover, individual opinions evolve during the decision making process and are influenced by the preferences of the other members of the group [3]. Hence, as some other authors have pointed out, group recommendation requires an interactive approach [6] [4]. A group RS should make the group members aware of each others' preferences and facilitate a negotiation to find an agreement, rather than computing over previously-acquired profiles [1].

One technology that seems appropriate for supporting the negotiation of recommendations is *critiquing* [5]. Critiquing recommenders accept input from users in the form of feature-specific feedback on presented items. For example, a user viewing a restaurant recommendation might say "Show me a restaurant like this one, but cheaper." This would be a critique of the "price" feature and directs the system to find similar restaurant in a lower price range. As this example shows, critiquing is a conversational activity that looks something like a negotiation, with the system proposing items and the user responding to them. In single-user applications, critiquing has been shown to be natural and useful mode of recommendation interaction [7].

Interactive multi-party critiquing (IMPC), the technique proposed in this paper, is an extension of the critiquing concept to a conversation between individuals. An IMPC interaction proceeds through multiple cycles of proposal, critique and counterproposal. Each counterproposal is accompanied by a critique that relates it to prior proposals.

We have implemented IMPC in a mobile phone application for group recommendation of restaurants. The interaction begins when a user proposes a restaurant and makes his preference known to the other party. If for some reason, the proposal is not acceptable, the responding user uses the system's critiquing interface to find an alternative. This counterproposal, along with the critique by which it was generated, is then presented to the originating party. The counterproposal can be critiqued in turn, and the decision making process continues.

This interaction provides two crucial features. One is that the counterproposal is likely to be close to the user's original proposal because it was generated by critiquing that proposal. So, the counterproposal has some likelihood of being

RecSys'11, October 23–27, 2011, Chicago, Illinois, USA.

acceptable to both parties. Second, the counterproposal is justified by the critique that the user applied: the originating party sees a succinct representation of the reason for the counterproposal. Together these features are intended to help the users to converge on a common solution, through mutual understanding of each other's preferences.

Our hypotheses therefore are that interactive multi-party critiquing is an *effective approach to group recommendations* and that, compared to simple feature-based search tools, critiquing *provides better support for group recommendation and decision making.*

In order to test these hypotheses, we developed two system prototypes that support of a pair of users in the exchange of restaurant proposals. In the first system, users exchange counterproposals, as described above, but the proposal are arrived at through feature-based search of the restaurant database, and no critique information is shared. In the second system, interactive multi-party critiquing is used as described above. In a live user experiment, we found that the variant with critiquing is more effective in helping the users finding an agreement, and it is also the one preferred by the users.

2. EXCHANGING RECOMMENDATIONS

Where2eat, our interactive multi-party critiquing application, is implemented using the Java2ME framework for mobile applications. Where2eat offers recommendations for 145 restaurants located in province of Bolzano, Italy. In Where2eat, each restaurant is represented as a vector of five (nominal) features: Price, Type, Location, Facilities, and Cuisine. For instance, the vector `<cheap, fast-food, Bolzano, {garden}, {italian, international}>` represents an inexpensive fast-food restaurant, located in Bolzano, with a garden, that serves Italian and international food.

The Where2eat decision process starts when a user initiates a session and ends when an agreement is found or the user to user conversation is aborted. In the initialization phase, User A decides to invite another party User B to dine together and starts a new session. The first step is to come up with an initial proposal. A feature-based search screen is provided to assist in this step: see Figure 1 left. The system generates a list of restaurants ranked by similarity to the user query, as shown in Figure 1 right. Once a restaurant has been selected, the application forwards User A's choice to User B along with a personal message. User B can then refuse the invitation entirely, accept the invitation as is, or move on to the proposal exchange phase.

The proposal exchange phase starts with User B searching for an alternative to User A's proposed restaurant to offer as a counterproposal. This search takes place through a critiquing interface: the user specifies new preferred values for certain restaurant features and the system attempts to find appropriate options. See Figure 2 left.

When a critique is given, the system uses the current proposal and the critiqued feature(s) to compute a new recommendation list. Each restaurant in the database receives a score based on how many restaurant features match the user's query. A feature value receives 5 points if it matches a critiqued feature and 1 if it matches a non-critiqued feature. The scores for all the features are added and the top N scoring restaurants are returned to the user. This scoring system ensures that options responsive to the user's critique are scored highest. When the user is satisfied with one of

Figure 1: Search and Results screenshots

the recommendations, it is sent as a counterproposal to the other party.

The system automatically generates an explanation for the counterproposal, formulated in first person, as shown in Figure 2 right. For example if the user expressed in the critique a preference for a restaurant with a garden, the generated explanation would be "I prefer a place with garden."

The counterproposal and its associated rationale then become input to the next stage of the group decision process. The two parties can continue the cycle of proposal, critique and counterproposal until an acceptable solution is found.

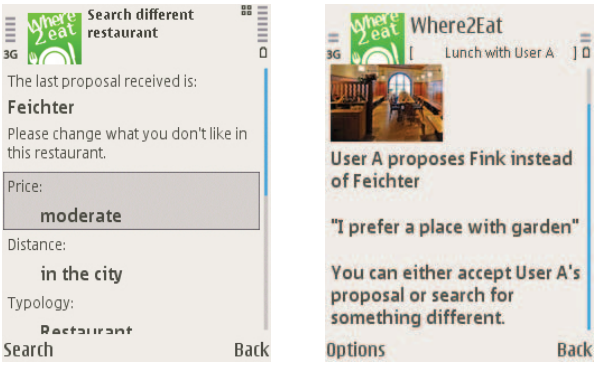

Figure 2: Critiquing and Explanation screenshots

3. EVALUATION AND RESULTS

In order to test our research hypotheses, we implemented two variants of Where2eat. The first variant supports critiquing for searching for new proposals, as described above. The second variant is similar, but critiquing is not available for the selection of counterproposals. Users only have the feature-based search capability used for finding the the initial restaurant proposal as shown in Figure 1.

These two variants were evaluated by 16 pairs of users (students and colleagues). Each subject received a Nokia N95 phone with the two applications installed and each pair was given the following task to complete: "Agree on a restaurant where to go together." Each pair had to agree on a common restaurant four times with different purposes for each run (e.g., a formal dinner or a fast lunch). Each subject invited the other peer (group member) two times, one with

each variant, choosing a different purpose each time. Half of the pairs started with the non-critiquing variant, the other half with the critique-based system. During the experiment two types of data were collected: subjective data (with a user survey) and objective data (with the application log).

3.1 Objective measurements

The results show that users of the critique-based application did come to consensus more quickly, although the effect is not large enough to be claimed as an advantage of the critique-based system (see Table 1). Users needed on average slightly fewer steps to reach an agreement, where a message exchanged and a search/critique submitted count each as one step. Also, slightly fewer iterations, i.e., restaurant proposals, were exchanged on average in the critiquing condition. If we look at how proposals were selected, we find that both systems were providing useful items: the rank of the selected item was between 1 and 2 in both cases. However, the results was slightly better in the critiquing condition. This shows that the users tended to prefer the results of the critiquing operation over feature-based search.

Table 1: Objective measurements

(Average quantity)	With critiquing	Without critiquing
Steps	5.875	6.063
Proposals exchanged	2.157	2.375
Ranking of selected rest.	1.05	1.46

More dramatic results were found when we examined the progress of each negotiation session. We expected that the differences between proposal and counterproposal would decrease until the parties agreed. This did happen in many sessions. However, we found there was a second pattern, in which feature differences actually increased during the negotiation, and the interaction ended with one of the parties "giving in" – accepting a proposal different from their own stated preferences.

As shown in Figure 3, the users of the critiquing system had many more sessions featuring converging proposals than diverging ones, where the opposite was true for the non-critiquing condition. We believe this dynamic contributes to the appeal of the critique-based system. Users of the other system were more likely to be end up capitulating to the preferences of the other party, and would not tend to think of the experience as successful.

3.2 Subjective evaluation

After using the system, users were asked to express a level of agreement to a list of statements on a 7-point Likert scale, from 1 (disagree) to 7 (agree). Some of the statements were taken from the Post-Study System Usability Questionnaire and some were added to better evaluate the specific characteristics of Where2eat. The users were asked to evaluate the system in general, to evaluate the variants separately and finally to directly compare the two variants.

Starting with the general system evaluation, we note that all the ratings range between 5.5 and 7. Strong positive sentiments were shown in statements S1: *It was simple to use the system*, S2: *The information provided by the system is clear*, S3: *It was easy to learn to use the system* and S6: *The features of the restaurants were clearly presented* – all between 6 and 6.4. The questions regarding the user interface (S4: *the on-screen information is helpful to solve the task* and S5: *the user interface is pleasant*) were rated around 5.8, showing that there is room for improvements in the graphic interface. There were divergent opinions regarding the last statement S7: *I would download and use the system if available*, rated 5.6. Some users were ready to adopt the system; those giving negative answers in many cases explained that they don't use mobile applications or smartphones in general. Finally some smartphone users, who are used to more sophisticated applications, said they would use Where2eat if some improvements were added, such as geolocation and mapping for users and restaurants.

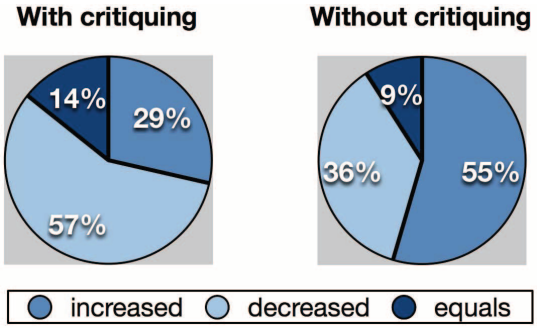

Figure 3: Differences between successive proposals

The agreements with the next statements were collected for the two variants independently. Again, positive feedback was given to all statements in general, although Where2eat with critiquing always scored higher: S8: *I am satisfied with the restaurants suggested by the system* (rated 6.16 with critiquing and 5.75 without); S9: *I am satisfied with the final restaurant choice* (rated 6.5 with critiquing and 6.21 without); S10: *Overall, I am satisfied with the system* (rated 6.56 with critiquing and 5.72 without, and this difference is significant for t-test, p=0,0008). Therefore we can say that users are more satisfied with the functionalities of Where2eat with critiquing, which is also confirmed by the high ratings of the two specific functionalities: *criticize the features of the received restaurants in order to get new offers* (rated 6.56) and *get an explanation for the proposal received* (rated 6.47).

Finally the users were asked to explicitly compare the two systems and the results are summarized in Figure 4. 91% of the users chose Where2eat with critiquing in Q2: *What system is more helpful to understand the preferences of other user?*, which shows that the critiquing interface and the explanations have the desired effect. On the other side there are different opinions about the question Q3: *What system is more helpful to express your needs to the other user?*, 69% of the users have chosen the variant with critiquing, where the user's preferences are explicitly showed in the explanation. The remaining 31% has chosen the variant without critiquing or judged the two systems equivalent, arguing that with a new search, the received proposal is ignored and the own preferences are imposed to the other user, which is another way to clearly express the individual needs. In Q4: *What system is more helpful to find an agreement?*, only 3% of the users have chosen the system without critiquing, which shows that almost all users recognize that the variant

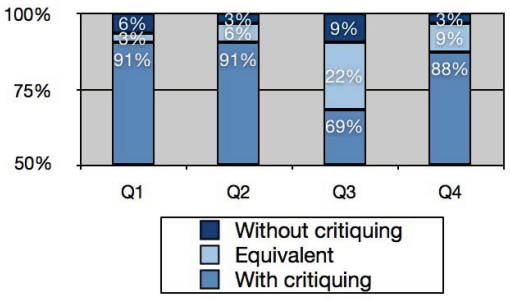

Figure 4: Comparison questions results

with critiquing helps more in the process of finding agreement. These results show that the users have understood the differences between the two systems and in Q1: *What system do you prefer?* the variant with critiquing was chosen by 91% of the testers.

Users who prefered the critiquing version of the system commented that they liked the ability to express explicitly what features they did not like in a restaurant proposal. When receiving a counterproposal, they found it useful to have an explanation of why their original proposal was not accepted. In general, they commented that critiquing made the interaction more transparent and conversational.

4. RELATED WORK

As mentioned in the Introduction, there are two main approaches to group recommendation [2]. The aggregation based method consists in making recommendations to the individual members of the group and then assembling the results (e.g., by averaging the predicted ratings for the group members and then re-ranking). The model based method consists in modeling the group as a single entity and providing recommendations directly for the group profile. Both of these approaches assume that a common recommendation can be based on existing static individual preference models. But this disregards the fact that in a real situation people's opinions evolve during a group discussion and that the individual preferences can be affected by group behavior [3].

One system that does incorporate an interactive approach to group recommendation is the CATS system [4]. This system is designed to help a group of friends planning a skiing vacation together. CATS considers both individual and group preferences to generate recommendations and dynamically collects feedback through critiquing. Users have the possibility to revisit a past recommendation after knowing that other members of the group are also willing to do so. So far this is the only example of a system that uses critiquing to update a multi-user model. In [6], in order to support group decision making, the authors presents an interface that enables the group members to see and reuse the preferences of the others.

In this paper we have suggested a completely different approach: to support a real-time recommendation exchange among the users by enabling them to make proposals whose selection was supported by critiquing. In this way the group has the possibility to interact and explore different possibilities, rather than seeking an "ideal" solution to be imposed by the recommender system.

5. CONCLUSIONS AND FUTURE WORK

This research introduces interactive multi-party critiquing for supporting group recommendation through exchanging recommendations. We have argued that, in order to enable a group decision making process, a system should allow users to exchange proposals and help them understand the preferences (motivations) that underlie these proposals. Because the process of critiquing requires users to incrementally articulate their preferences, it is well-suited to support this type of process.

We have described our implementation of IMPC in a mobile recommender for restaurants and shown through a small-scale study that the approach has benefits both in terms of decision-making effectiveness and in decision results. The critiquing-based system enabled the emergence of group consensus, as the exchanged proposals converge to the final selection, and was preferred by users overall. Hence, the critiquing system helped users converge on a common solution. The non-critiquing system tended to encourage "all or nothing" results more likely to satisfy one user than the other.

This work is very suggestive of the potential of IMPC. We are looking at how the technique might be extended to larger groups of users and to recommend other types of items.

6. REFERENCES

[1] S. R. de M. Queiroz and F. de A. T. de Carvalho. Making collaborative group recommendations based on modal symbolic data. In *Advances in Artificial Intelligence - SBIA 2004, 17th Brazilian Symposium on Artificial Intelligence, São Luis, Maranhão, Brazil, September 29 - October 1, 2004, Proceedings*, pages 307–316, 2004.

[2] A. Jameson and B. Smyth. Recommendation to groups. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *The Adaptive Web*, volume 4321 of *Lecture Notes in Computer Science*, pages 596–627. Springer, 2007.

[3] J. Masthoff. Group recommender systems: Combining individual models. In F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, editors, *Recommender Systems Handbook*, pages 677–702. Springer, 2011.

[4] K. McCarthy, M. Salamó, L. Coyle, L. McGinty, B. Smyth, and P. Nixon. Cats: A synchronous approach to collaborative group recommendation. In *Proceedings of the Nineteenth International Florida Artificial Intelligence Research Society Conference*, pages 86–91, Melbourne Beach, Florida, USA, 2006.

[5] L. McGinty and J. Reilly. On the evolution of critiquing recommenders. In F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, editors, *Recommender Systems Handbook*, pages 419–453. Springer, 2011.

[6] C. Plua and A. Jameson. Collaborative preference elicitation in a group travel recommender system. In *Proceedings of the AH 2002 Workshop on Recommendation and Personalization in eCommerce*, pages 148–154, Malaga, Spain, 2002.

[7] P. Viappiani, B. Faltings, and P. Pu. Preference-based search using example-critiquing with suggestions. *Journal of Artificial Intelligence Research*, 27:265–503, 2006.