Towards more Conversational and Collaborative Recommender Systems

Giuseppe Carenini, Jocelyn Smith, David Poole
Department of Computer Science
University of British Columbia
2366 Main Mall, Vancouver, B.C. Canada V6T 1Z4
+1 604 822-5109
carenini,poole,jdsmith@cs.ubc.ca

ABSTRACT
Current recommender systems, based on collaborative filtering, implement a rather limited model of interaction. These systems intelligently elicit information from a user only during the initial registration phase. Furthermore, users tend to collaborate only indirectly. We believe there are several unexplored opportunities in which information can be effectively elicited from users by making the underlying interaction model more conversational and collaborative. In this paper, we propose a set of techniques to intelligently select what information to elicit from the user in situations in which the user may be particularly motivated to provide such information. We argue that the resulting interaction improves the user experience. We conclude by reporting results of an offline experiment in which we compare the influence of different elicitation techniques on both the accuracy of the system's predictions and the user's effort.

Keywords
Recommender systems, Collaborative filtering, Preference elicitation, Mixed-initiative interaction, Motivation

INTRODUCTION

You are thinking of going to see a movie but you are not sure you would enjoy it. Probably, you would ask your friends for advice. They might have seen it and they should be willing to give you their rating for it. Alternatively, imagine that an acquaintance recommends you go see a movie that you do not think you would enjoy. In this case, you would be the one willing to provide information to help your friend make better recommendations in the future. Current recommender systems do not allow this type of interactive process to occur between the system and its users. Our goal is to create a system that is more conversational and allows the users and the system to "work together" (collaborate) to improve the system's recommendations and the user experience.

Recommenders are computer systems designed to help people find preferred items within a very large set of available alternatives (e.g., movies, books, news). In order to provide effective, tailored advice (i.e., to predict whether the user will like/dislike a particular item) recommenders can make use of three different sources of information about their users: (i) user demographics (e.g., user's age, gender), (ii) user preferences about features of the items (e.g., the movie director) and (iii) user ratings on experienced items (e.g., previously seen movies). All these sources of information can be effectively combined (e.g., [1]). However, in this paper we focus on pure *collaborative filtering* (CF) [2], a set of techniques for recommending that only relies on third source of information: users' explicit ratings for items they have experienced.

The goal of CF is to make predictions about the user preferences; therefore, its acceptance as a viable technology clearly depends on how accurate the predictions are. The accuracy of CF arguably depends on at least two key factors: (i) the effectiveness of the algorithm that combines the available ratings to generate predictions (ii) the density of the set of available ratings. Thus, research aimed at increasing CF prediction accuracy should focus on both developing algorithms that effectively combine the available ratings, as well as, devising techniques to elicit as many ratings as possible from users. Nevertheless, most CF research so far has been devoted to investigating only the first aspect of the problem. For instance, see [3] for a review and an empirical comparison of the accuracy of different CF algorithms for combining ratings, and see [4] and [5] for more recent and more accurate algorithms.

Only very recently has there been a growing interest in addressing the other side of the problem, namely, in devising techniques to effectively elicit ratings from the user. More specifically, the goal here is to maximize the number of elicited ratings, minimize the number of questions asked to obtain these ratings and consequently minimize the user time and effort. So far, elicitation techniques for CF have been studied only in the context of solving the *new user problem*, which is the problem (faced by any recommender system) of acquiring information from a new user in order to generate personalized recommendations. A clear review of previous work and the most recent and comprehensive approach to the new user problem in CF can be found in [6]. Several techniques for eliciting ratings from a new user are proposed and compared with respect to measures related to the quality of the user experience.

It is rather uncontroversial that the initial registration phase is an appropriate time to elicit as many ratings as possible from a new user. However, we claim that a model of interaction based on the assumption that the recommender is actively and intelligently eliciting ratings only during the initial registration is limited from the perspective of human-computer interaction and computer-support of collaboration. In such a model, no collaborative dialogue occurs between the user and the system. Once the registration phase is concluded, the flow of information between the user and the system is asynchronous and only occurs through two rather independent communication channels: the system generates recommendations and users autonomously provide new ratings whenever they feel like it. Furthermore, the collaboration among all users is indirect. A user collaborates with other users only because when she provides information about herself, it becomes easier for the system to generate more accurate predictions for other users.

We believe there are several unexplored opportunities in which ratings can be effectively elicited from the user by making the underlying interaction model more conversational and collaborative. We identify some situations in which a user may be particularly motivated to provide more ratings. We propose that the system should exploit these situations by soliciting ratings for items that are intelligently selected for the specific situation. The resulting interaction will both increase the accuracy of the system's predictions and arguably improve the user experience. And this interaction should be acceptable to users, because, as recent empirical studies on the usability of commercial recommenders indicate, users do not mind rating more items to receive more accurate recommendations [7].

In the remainder of the paper, we will first introduce CF basic principles and terminology. Then, we compare the interaction model currently assumed for recommender systems with our new proposal. A discussion of elicitation techniques will follow. We conclude with a detailed analysis of the results of a preliminary empirical evaluation of the proposed techniques and a discussion of future work.

COLLABORATIVE FILTERING

In essence, a recommender system predicts how much a user will like each item in the datasets (e.g., movies, books, news). And then, it recommends to a particular user the item(s) it predicts that this user will like the most.

One major class of recommender systems is based on collaborative filtering (CF). In making predictions, CF relies only on users' explicit ratings for items they have experienced. CF uses the information in a large database of users' ratings for items to make predictions for ratings corresponding to a particular item and user, for which there is not yet a recorded rating. More specifically, the input of a CF system is a large, partially populated rating matrix, in which the rows correspond to the users, the columns correspond to the items and the value in the cell *(i,j)* (when filled in) corresponds to the rating of the user *i* for the item *j*. Breese et al. [3] identify two basic subclasses of algorithms for CF: Model Based and Memory Based. Model based algorithms build a model of the data and use the model for prediction, whereas memory based algorithms do not build a model, but use the data directly for prediction. In our study, we used a memory based *Correlation* (CR) algorithm that Breese et al. found in their analysis to be one of the best performers. Memory based algorithms work under the assumption that a user will like a particular item if similar/dissimilar users (according to some measure of similarity between users' ratings) also liked/disliked the item in question. In CR, the correlation between the ratings of two users is the measure of their similarity. In practice, in order to predict u_i's rating for it_j, this algorithm first computes the correlation between u_i's ratings and the ratings of each of the other users in the database who rated it_j (these users are called the neighbors of u_i for it_j). Then, the correlation between u_i and each neighbor determines how influential the neighbor's rating of it_j will be on the prediction for u_i's rating for it_j.

We have adopted the CR algorithm in our study, because it provides a very intuitive and practical notion of similarity between users. As described later, this notion plays a key role in our techniques for eliciting new ratings from users.

INTERACTION MODELS FOR RATING ELICITATION

Eliciting ratings from users is a critical task in CF, because the accuracy of recommendations depends not only on the quality of the prediction algorithm, but also on the number of ratings available (i.e., how populated the input rating matrix is)[1].

In current CF recommenders, rating elicitation relies on a fairly inflexible interaction model, that we call the Standard Model. In this section we propose a new model of interaction for eliciting ratings that is more conversational and collaborative.

Standard Interaction Model: in this model intelligent elicitation of ratings only occurs at registration time [6]. When a new user registers, she is presented with lists of items to rate until she has rated a sufficient number of items.

[1] The quality of the ratings is also important, but we do not deal with this aspect in this paper.

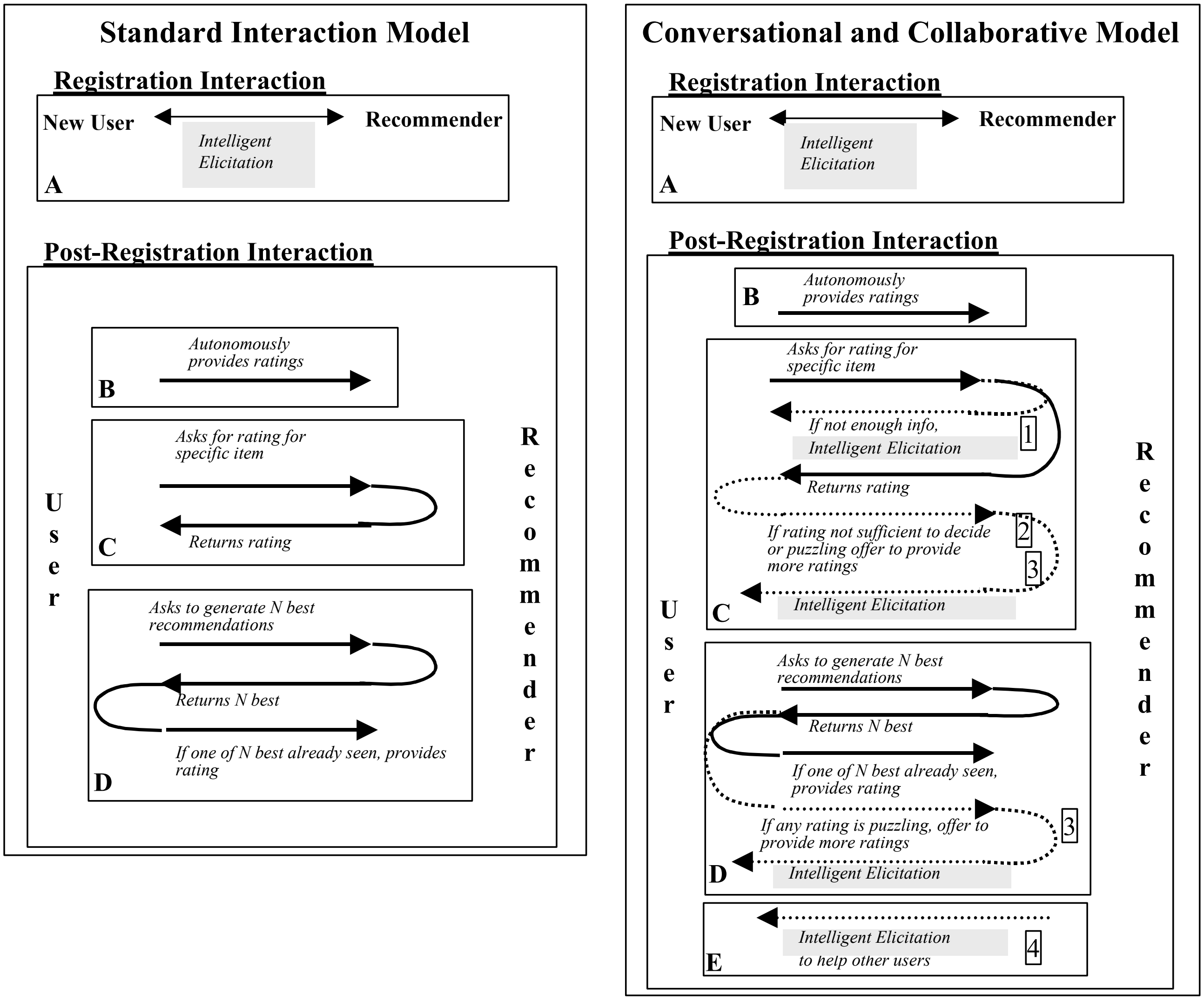


Figure 1 Interaction Models for Rating Elicitation

After that, the system and the user communicate in an independent and asynchronous way. On the one hand, the system, when asked, provides recommendations. On the other, the user provides new ratings whenever she wants. The items she selects to rate are either ones she seeks out to rate, items the system has recommended but that she's already seen, or items the system randomly selects.

Conversational and Collaborative Model (CC) - While still allowing the user to rate items in the traditional way, we propose to also exploit situations where the user has additional *motivation* to supply more ratings. In these situations, which we discuss and justify below, the system can ask the user if she would like to rate more items in such a way as to make the benefit of doing so clear to the user. The user can then make the decision of whether or not to enter into such a dialogue with the system. Thus, the control is still in the user's hands, but a sense of co-operation between the user and the system is also developed.

Consider the following three facts that are consistent with most theories of human motivation [8], [9]. People are more willing to provide information if they perceive that their effort will: (i) contribute to the achievement of their immediate goals or the goals of their peers, (ii) contribute to the reduction of an error they care about or (iii) help other people who may be likely to reciprocate in the future.

In light of these three facts, we have identified four situations in which the user of a recommender system (based on CF) may be willing to provide more ratings.

(1) The user asks for a rating for an item she heard about and is interested in. The system assesses that it does not have enough information about the user to generate a sufficiently accurate prediction for that item. The system asks the user for help, by asking her to provide more ratings.

(2) As in the previous case, the user asks for a rating for an item she heard about and is interested in. The system

predicts an average rating (i.e., not too bad, not too good). To decide whether to select that item (e.g., whether to watch a particular movie), the user is willing to put in additional effort (i.e., provide more ratings) to get a better supported recommendation.

(3) The user is puzzled by a recommendation (i.e., a predicted rating), because: she had experienced the item and her rating for that item would be significantly different; or she has not experienced the item, but she believes her rating would be significantly different (e.g., because she saw the trailer of the movie).

(4) If the user rates certain items, the accuracy of the ratings for other users can be improved. And if she helps them, they may reciprocate in the future.

Figure 1 shows a diagram that summarizes and compares the Standard and the CC interaction models (left and right respectively). In the figure, a conversational unit (i.e., a sequence of related exchanges between the user and the system) is enclosed in a box and indexed by capital letters (e.g., A, B). Within a unit, arrows specify the conversation flow. The starting point of an arrow indicates who is making the contribution (i.e., the user or the recommender) and an arrow branching indicates a conditional branching. For instance, in the CC model unit D, first it is the user who asks for recommendations. Next, the recommender returns the N best items. After that, if the user has already experienced one of the items, she can provide her rating for that item. And, if the user is puzzled by one of the ratings she can offer to provide more ratings.

In general, while comparing the two models in Figure1, notice that, although the Standard and CC models are the same with respect to the registration of a new user (upper box), they are very different as far as the post-registration interaction is concerned (lower boxes). More specifically, not only does the CC model comprise an additional conversational unit (i.e., unit E), but units C and D also specify more complex interactions in CC (all additional possible exchanges are displayed as dashed arrows). Furthermore, as highlighted in gray in the figure, in CC the system has four additional opportunities for further intelligent rating elicitation. These opportunities closely correspond to the situations we itemized above, in which the user may be more motivated to provide more ratings. In the figure, boxed numbers indicate this correspondence.

Of course, the obvious advantage of CC is that by exploiting more opportunities for eliciting rating, it should be able to generate, on average, more accurate predictions (in particular when the elicited ratings are intelligently selected). However, besides this obvious benefit, we argue that the CC model may provide several additional advantages in term of user experience:

(i) The elicitation of ratings is performed in situations in which the user is more motivated to provide information. And this may improve the user experience.

(ii) Even assuming that the same amount of elicitation is performed both in the Standard and in the CC models, in CC the elicitation of ratings is decomposed into smaller chunks. Since the elicitation of ratings is a repetitive and tedious task this decomposition may also improve the user experience.

(iii) In CC, the system is helping the user by providing recommendations, but the system is also asking the user for help whenever more information is needed to generate a more accurate prediction (situation #1). As described in [10] Chp. 14, a more balanced user-system interaction, based on mutually giving and receiving help, is an important factor in fostering a user's view of the system as a teammate. When this is achieved, the user will like the system more, will be more willing to cooperate with the system, and will agree more with the system. These are all critical aspects in determining the user experience.

(vi) In situation #4, the user is asked for ratings to help other users get more accurate recommendations. If users comply with these requests, this may create a feeling of interdependence and cooperation among users. In a later section, we will discuss how user compliance can be stimulated and as a result, a sense of belonging to an online community be fostered.

To summarize, we have identified four situations in which users may be willing to put extra effort by providing more ratings and we have argued four reasons why eliciting ratings in these situations might improve the user experience. We turn now our attention to the question of how the recommender can intelligently select what ratings to elicit from the user in those situations.

TECHNIQUES FOR ITEM-FOCUSED PREFERENCE ELICITATION

Our techniques to elicit ratings from a user in the CC model post-registration interactions are variations of strategies, recently proposed in the literature, for eliciting ratings from new users at registration time [6]. We will first describe these strategies for new users and then present our techniques for post-registration elicitation.

New user elicitation strategies

Five strategies are presented and evaluated in [6]: random, entropy, popularity, balanced (popularity and entropy) and item-item. Before examining the strategies in detail, notice that their evaluation considered two basic dimensions related to the user experience:

User effort: an effective elicitation technique should ask for ratings that the user is likely to be able to provide. If this is not the case, the user will be forced to go through a large number of items she cannot rate, a process that will be tedious and frustrating.

Recommendation accuracy: not all ratings are equivalent, some ratings provide more valuable information for prediction than others. An effective elicitation technique should acquire ratings that allow the recommender to maximally improve its predictions.

In general, except for the random strategy which is used as a baseline, all the strategies presented in [6] attempt to either optimize one of these two dimensions or find an acceptable trade-off between them. Here are the details of each strategy.

Random (baseline): select items to present randomly with uniform probability from the set of all available items. No intelligent analysis on the rating matrix is performed.

Popularity (Pop): present items in descending order of popularity. The popularity of an item corresponds to the number of ratings for that item in the rating matrix (see section on CF for a definition of the rating matrix). Clearly, this strategy tries to optimize the user effort dimension. A user is more likely to have experienced more popular movies and therefore be able to rate them. However, popular movies are typically liked, so knowing a user's rating for a popular movie does not provide much valuable information to the recommender. Thus, this strategy is not expected to fare well with respect to the dimension of recommendation accuracy.

Entropy(Ent): intuitively, a rating for an item that has received very diverse ratings will give more information than a rating for an item that everybody likes (or dislikes). The entropy on an item is a formal measure of the diversity of the item ratings in the rating matrix. In the entropy strategy items are presented in descending order of entropy. With respect to the user experience, since item entropy is an estimate of the information its rating will provide to the recommender, this strategy tries to optimize that recommendation accuracy dimension. However it is not expected to fare well with respect to user effort.

*(log Popularity)*Entropy (PopEnt)*: this strategy is an attempt to find a reasonable trade-off between the benefits of the popularity and entropy strategies. Items are presented in descending order of the product of the log of their popularity times their entropy (the log of the popularity is used to avoid popularity completely dominating the entropy). This strategy is expected to perform reasonably well on both dimensions of user effort and recommendation accuracy.

Item-item personalized: present items that are similar to the ones the new user has already rated. Obviously, this strategy must be initialized by running one of the above strategies to obtain an initial set of rating for the new user. With respect to the user experience, since people tend to experience similar items (e.g., tend to go to similar movies), this strategy tries to optimize user effort, with a possible loss in recommendation accuracy.

An online user study to evaluate these strategies [6] confirmed most of the expectations we have mentioned. In particular, as summarized in Table 1, a variation of the random strategy (Classique) performs poorly on both dimensions of the user experience. Pop and item-item require the least user effort, with Pop being much better for accuracy between these two. The PopEnt strategy turned out to be the best for accuracy, without requiring an unreasonable user effort[2].

Strategy	*User effort*	*Accuracy*
Classique	✮	✮ ✮
Pop	✮ ✮ ✮ ✮ ✮	✮ ✮ ✮ ✮
PopEnt	✮ ✮ ✮	✮ ✮ ✮ ✮ ✮
Item-item	✮ ✮ ✮ ✮ ✮	✮ ✮

Table 1 Evaluation of elicitation strategies for new users [6]

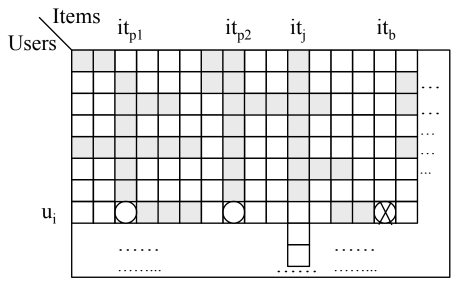
Item-focused elicitation techniques

The key difference between our elicitation techniques and the new user strategies is that while the new user strategies elicit ratings to improve prediction accuracy on the whole set of unknown ratings, our techniques elicit ratings to improve prediction on a specific item. For this reason they are called item-focused.

Let's now briefly go back to the CC model. Notice that the common feature of all the situations in which the user is motivated to provide more ratings is that there is a problem with the recommender's prediction for a particular item. In situation #1, the system does not have enough information to make a confident prediction, whereas in situation #2, the prediction is not sufficiently informative. In situation #3, the user is puzzled by the prediction. Finally, in situation #4, the prediction for another user is problematic and the current user is asked for help. Our techniques are designed to be effective precisely in these situations, in which the goal is to elicit ratings to improve the prediction on a specific item. Item-focused techniques are essentially of two different types: techniques to select what rating to elicit from the current user; and a technique to select what rating to elicit from other users (who will therefore find themselves in situation #4). We call techniques of this second type collaborative. Techniques of the first kind are variations of the new user techniques previously described. In particular, we only adapted the Pop and the PopEnt strategies, because overall they dominate the other strategies (see Table 1). In essence, the key distinctive aspect of our techniques is that both the popularity and entropy of items are not computed on the whole rating matrix, but only with respect to the rating sub-matrix including the current user and her neighbors for the item in focus (other users in the database who rated that item).

---

[2] In an offline experiment the entropy strategy performed extremely poorly on both dimensions of user experience. Hence, it was not evaluated in the online experiment.



In the rating matrix above:
- a cell is gray if the rating for the corresponding user/item pair is known
- users u_0-u_{i-1} are the neighbors of u_i with respect to it_j
- the prediction of (u_i, it_j) would be maximally influenced by (u_i, it_{p1}) and (u_i, it_{p2}) (it_{p1} and it_{p2} are the most popular items among the neighbors), but not at all by (u_i, it_b) (no neighbor rated it_b)

Figure 2 Rationale for item-focused popularity

To clarify the rationale behind our approach, let's review the basics of CR-based CF. As explained earlier, a recommender, in order to predict u_i's rating for it_j, first computes the correlation between u_i's ratings and the ratings of each of the other users in the database who rated it_j (i.e., the neighbors of u_i for it_j). Then, the correlation between u_i and each neighbor determines how influential the neighbor's rating of it_j will be on the prediction for u_i's rating for it_j. Thus, if we want to ask u_i for ratings that are going to maximally influence this computation, we have to ask for ratings that are popular among the neighbors of u_i, because these ratings will be involved in the computation of most correlations (see Figure 2). Also, since the ratings used in the computation are only the ones of u_i and her neighbors, it seems reasonable to modify the balanced strategy so that the item entropy is computed on this subset of the rating matrix.

As for the collaborative technique, to select what ratings to elicit from other users, we propose to ask users who are strongly correlated with u_I to rate it_j. The rationale is that by expanding the set of neighbors with users who are strongly correlated with u_i, we will get information that will be maximally influential in computing a prediction on it_j for u_i.

In summary, our item-focused techniques try to elicit ratings to increase the prediction accuracy for the item of interest, hopefully limiting the user effort. And this is done by intelligently eliciting ratings either from the user for whom the prediction is computed, or in a collaborative way from other users.

EVALUATION

Rashid et al [6] propose and successfully apply a three-step methodology for the design and evaluation of techniques for eliciting preferences in recommender systems. First, perform a semi-formal analysis of the elicitation problem and develop of corresponding elicitation techniques; next, perform a set of offline experiments on previously collected data to quickly focus on the most promising techniques. And finally, run online user studies to control for biases in the data used in the offline studies and, more importantly, to verify the acceptability and effectiveness of the techniques in practice.

Naturally, in order to perform online user studies, you need access to a working recommender with a sufficiently large community of users. Since we currently do not have access to such a system, in the evaluation of our interaction model and associated techniques, we have so far only applied the first two steps of this methodology: design and offline evaluation. In the previous section we have described the design of our item-focused elicitation techniques, in the following we will discuss an offline evaluation of these techniques.

For our study, we used a portion of the publicly available Eachmovie dataset[3] (the same portion used in [3]). Eachmovie is a recommender of movies based on collaborative filtering, in which users rate movies and receive recommendations on a scale 0-5 (where 0 means an awful movie, while 5 means an excellent movie).

Similarly to [6], in order to ensure that we had enough data for testing the elicitation strategies, we only considered users who rated at least 200 movies. This left us with a 1,813x1,618 rating matrix (i.e., 1,813 users and 1,618 items) populated with 518,197 ratings. Hence, each user has expressed on average 286 ratings.

Again, following [6], we compare elicitation techniques according to two basic dimensions related to the user experience (that we discussed earlier): user effort and recommendation accuracy. User effort is minimized the more items the user is able to rate given a certain number of requested ratings, while accuracy is maximized the more the error in the system's predictions (after elicitation) is reduced. As a measure of error, we use the standard Means Absolute Error (MAE), which is computed as the means of the absolute differences between the system predictions and the corresponding actual user ratings.

In order to simulate and test elicitation techniques on a particular user, her known ratings are split into two subsets: a subset that we assume is initially known by the system (K), and another subset that is hidden from the system (H). Then, as an elicitation technique is applied, the user "can provide" the system with a rating if it is in H. More specifically, to evaluate an item-focused technique to elicit ratings from the current user, first an item it_f (whose rating is in H) is selected as the one on which to focus the elicitation. Then, if we call R the set of all ratings known by the system, an initial prediction is computed for it_f by using the ratings in K' = R - H. After that, the elicitation technique is applied by asking the user for n ratings; and the user provides a rating if it is in H. Let's call the set of all ratings that the system elicits E (which is a subset of H).

[3] See *http://research.compaq.com/SRC/eachmovie/* for more information.

Once elicitation is completed, the two dimensions of elicitation effectiveness are measured as follows: user effort depends on |E| (the number of items that the user could rate out of the requested n), the higher |E| is the better; the accuracy is measured as the reduction in the error between the prediction for it_f using K', and the prediction for i_f using $K' \cup E$.

To evaluate a collaborative item-focused technique that elicits ratings from other users, a similar process is applied to the neighbors (N) of the current user u_i for it_f. A random 70% of their ratings for it_f (HC) is hidden from the system (leaving only a 30% of known ratings (KC) known to the system). Next, the elicitation technique is applied to the neighbors most strongly correlated with u_i and they provide a rating for it_f if it is in HC (let's call the elicited set EC). Once elicitation is completed, accuracy is measured as the reduction in the error between the prediction for i_f using KC' = R - HC, and the prediction for it_f using $KC' \cup EC$[4]. When both types of techniques are applied, accuracy is measured as the reduction in the error between the prediction for i_f using $R - (H \cup HC)$, and the prediction for it_f using $[R - (H \cup HC)] \cup E \cup EC$.

Req. Ratings	30		50		70	
Strategies	Change in MAE	Obtained ratings	Change in MAE	Obtained ratings	Change in MAE	Obtained ratings
Random	0.03	6.3	0.04	6.3	0.05	9.3
Unfocused Pop	0.02	7.4	0.03	12.1	0.03	12.7
Unfocused Pop*Ent	0.04	10.5	0.04	13.1	0.05	14.2
Focused Pop	0.06	12.5	0.08	21.6	0.1	30.4
Focused Pop*Ent	0.06	12.8	0.09	22	0.1	30.4
Collaborative Focused(Pop*Ent)	0.08	46	0.1	46	0.11	46

Table 2 Evaluation Results

We have examined what we measured in our study and how we measured it, let's now see in detail how the study was organized. We split the users in our Eachmovie dataset into two groups: half the users were used as training data and the others to test the techniques. Since our item-focused techniques are intended for relatively new users, for each user in the test set we assumed that she had already gone through the initial elicitation at registration (unit A in Figure1) based on the popularity strategy, and 50 ratings had been elicited. All the other ratings known about the user were hidden, as described before. So, when an elicitation technique is applied to a test user, K includes these 50 movies and K' is $K \cup$ the set of all the ratings in the training data. When a collaborative technique (i.e., that elicits ratings from other users) was tested, the neighbors and their rating for i_f came from the training data.

[4] In this case the user effort measure does not make sense, because each rating is elicited form a different user.

Table 2 shows the results of our study. We considered three unfocused strategies as baselines (Random, Pop and PopEnt), the two focused Pop and PopEnt, and a combination of focused PopEnt and collaborative. In combining focused PopEnt and collaborative, we tested them in sequence. First, ratings are elicited from the user according to focused PopEnt. Then, by using the newly obtained ratings, a list of the 200 closest neighbors is computed and those neighbors are asked for their input. We tested all the strategies with n (the number of ratings requested to the current user) equal to 30, 50 and 70. For each strategy we report the reduction in MAE and the obtained ratings (for collaborative, we report ratings obtained from other users).

As shown in Table 2, there are no differences between the two focused strategies (except for a small difference in accuracy when n=50). However, focused strategies are consistently better than unfocused ones and when combined with the collaborative strategy the benefit is even more pronounced (all measures reported are averages of multiple runs and all difference are statistically significant). Notice that for item-focused techniques, we get MAE reductions in the range 0.06–0.11. This may seem negligible given the 0-5 interval for ratings. However, consider that the MAE on the testing set before applying item-focused elicitation is on average 1.03. Therefore, what we obtained is a 11% reduction of that error. Also, similar results (modulo the size of the dataset) were obtained for new user elicitation strategies in [6]. Furthermore, MAE, although commonly adopted, is considered a rather rough and somehow unrealistic measure of error; thus the benefits of item-focused techniques might be more substantial on a more realistic measure. We are currently investigating a more realistic measure based on utility theory. Finally, in this experiment we have performed only one item-focused elicitation on one user in each run. We expect that if item-focused techniques were applied in a working recommender on several users, as more and more multiple parallel item-focused elicitations are performed over time, they would together keep the rating matrix sufficiently populated and effectively contribute to increasing prediction accuracy and reduce user effort (in particular if new items are added over time, e.g., new movies come out).

ADDITIONAL BENEFITS OF ITEM-FOCUSED TECHNIQUES

Our preliminary offline evaluation indicates that item-focused techniques are beneficial in terms of prediction accuracy and user effort. However, as we have argued in a previous section, item-focused elicitation in the CC model may provide several additional benefits: ratings are elicited in situations in which the user is more motivated; elicitation (a possibly tedious task) is decomposed into smaller chunks; a user's view of the system as a teammate may be fostered; and finally, a feeling of interdependence and cooperation among users may be created.

This final advantage deserves particular attention, because it may play an important role in contributing to the creation of a successful online community [11]. Remember that when a collaborative item-focused elicitation technique is applied, users are asked for ratings to help other users. However, sociology tells us that people do not help each other every time they are given a chance. Cooperation among people occurs only if three conditions are met [11]. First, the probability that two individuals meet again must be high. Second, individuals must be able to identify each other. And third, individuals must have information about how others have behaved in the past.

It is an open question whether these conditions can be enforced among users of recommender systems. The second condition could be easily satisfied if users were willing to use pseudonyms (something currently under investigation [12]). And we speculate that item-focused collaboration may contribute to the establishment of the other two conditions for cooperation. When the collaborative elicitation technique is applied, it potentially generates a large number of opportunities for users to "meet" and help each other. Also, if the system keeps track of who has provided ratings for whom, this "institutional memory" could be used by the system to stimulate further collaboration. For instance, the system could elicit ratings more effectively by mentioning that the users who need those ratings helped the current user several times in the past. Or, alternatively the system could mention that a certain rating may help new users who did not have a chance yet to contribute to the community, but who may do so in the future, especially if they are helped at the beginning.

CONCLUSIONS AND FUTURE WORK

The effectiveness of recommenders critically depends on the amount of information they manage to collect about their users. We claim that eliciting ratings can be turned from a burden into an asset by making the elicitation process more conversational and collaborative. Essentially, it is crucial to ask for ratings in situations in which users are motivated to provide them either to help themselves, to help the system or to help other users. We propose a set of elicitation techniques appropriate for those situations, and an offline empirical evaluation indicates that applying these techniques is beneficial in terms of prediction accuracy and user effort.

The obvious next step for our research is to complete our investigation by performing an online user study to verify the acceptability and effectiveness of our approach in practice. We plan to do this by establishing a collaboration with a research group that already has a working recommender with a sufficiently large community of users.

Other areas we intend to explore include investigating the performance of item-focused techniques with more realistic measures of prediction error, as well as devise item-focused elicitation techniques for recommenders not exclusively based on collaborative filtering.

REFERENCES

1. Popescul, A., et al. *Probabilistic Models for Unified Collaborative and Content-Based Recommendation in Sparse-Data Environments*. in *Uncertainty in Artificial Intelligence*. 2001.
2. Resnick, P., et al. *GroupLens: An open architecture for collaborative filtering of netnews*. in *Proceedings of the 1994 Computer Supported Collaborative Work Conference*. 1994.
3. Breese, J., D. Heckerman, and C. Kadie. *Empirical Analysis of Predictive Algorithms for Collaborative Filtering*. in *Uncertainty in Artificial Intelligence*. 1998.
4. Sharma, R. and D. Poole. *Symmetric Collaborative Filtering Using the Noisy Sensor Model*. in *Uncertainty in Artificial Intelligence: Proceedings of the Seventeenth Conference (UAI-01)*. 2001. San Francisco, CA,.
5. Pennock, D.M., *et al. Collaborative filtering by personality diagnosis: A hybrid memory- and model-based approach*. in *Uncertainty in Artificial Intelligence*. 2000.
6. Rashid, A.M., et al. *Getting to Know You: Learning New User Preferences in Recommender Systems*. in *Intelligent User Interfaces*. 2002.
7. Swearingen, K. and R. Sinha. *Beyond Algorithms: An HCI Perspective on Recommender Systems*. in *SIGIR workshop on Recommender Systems*. 2001. New Orleans, LA - USA.
8. Edwards, D.C., *Motivation & Emotion*. 1999: SAGE Publications.
9. Weiner, B., *Human Motivation: Metaphors, Theories and Research*. 1992: SAGE Publications.
10. Reeves, b. and C. Nass, *The Media Equation*. 1996: Cambridge University Press.
11. Preece, J., *Online Communities: Designing Usability, Supporting Sociability*. 2000: Wiley.
12. Konstan, J.A. *Building Community Partecipation in Reccomender Systems*. in *Workshop on Online communities: Supporting Sociability, Designing Usability*. 2000. HCI Lab University of Maryland.