Diversity-Enhanced Conversational Collaborative Recommendations

Derek Bridge and John Paul Kelly

Department of Computer Science,
University College, Cork
d.bridge@cs.ucc.ie/jpk2@student.cs.ucc.ie

Abstract. In conversational collaborative recommender systems, user feedback influences the recommendations. We summarise the seminal work in this field [5] and make precise a variant in which the likes and dislikes of other users in the system are distinguished when matching against the active user's short-term positive and negative profiles. But the major innovation that we report is our mechanism for enhancing the diversity of the recommendations made by collaborative recommenders. Significantly, we increase diversity using collaborative data only. In our experiments with diversity-enhanced recommendations, users find target items in many fewer recommendation cycles.

1 Introduction

Recommender systems suggest products, services or information sources to their users. They differ in the way they find the items they recommend:

Content-based systems: The system stores a description of each available item. A user describes the item that she wants as a query or she describes the kinds of items that she likes as entries in a user profile. The system compares the user's descriptions against the store of item descriptions and recommends items that match.

Collaborative systems: Item descriptions are not used. A user's profile stores user opinions against item identifiers. The system compares other users with the active user and recommends items that were liked by users whose profiles are similar to the active user's profile.

Recommender systems differ also by the extent to which they engage in dialogue with the user:

Single-shot systems: In response to a user request (and, where appropriate, submission of a user query), the system delivers a set of recommendations to the user. Each request is treated independently of previous ones.

Conversational systems: Users elaborate their requirements over the course of an extended dialogue. On receiving a set of recommendations, the user might refine her query; or she might supply feedback on the recommended items. Her feedback influences the next set of recommendations.

Table 1. A ratings matrix

	Ann	Bob	Col	Deb	Edd	Flo
Cape Fear	⊥	⊥	3	5	5	5
Naked Gun	3	2	⊥	2	4	⊥
Aliens	⊥	5	⊥	⊥	2	4
Taxi Driver	⊥	⊥	3	4	3	⊥

Conversational systems can more easily adapt their recommendations to the user's short-term interests. By dint of mood changes or other special circumstances, short-term interests may not coincide with long-term interests.

There is a mature body of research on conversational *content-based* systems, but research into *collaborative* systems has focused on single-shot recommenders. The work of Rafter & Smyth [5] is a recent exception. In Sects. 3, 4 and 5, we describe conversational collaborative recommenders of increasing sophistication, starting from the one in [5]. The results of an empirical comparison are reported in Sect. 6. But first, in Sect. 2, we summarise the operation of collaborative recommenders.

2 Collaborative Recommenders

In a collaborative recommender, given m items, $I = \{i : 1 \ldots m\}$, and n users, $U = \{u : 1 \ldots n\}$, preferences are represented using a $m \times n$ matrix of ratings $r_{i,u}$. Note that it is possible and common that $r_{i,u} = \bot$, signalling that the user has not yet rated that item. An example of a ratings matrix for movies is shown as Table 1. Each column in the matrix is a user's long-term profile. We will write u^{LT} for the item identifiers that have non-$\bot$ ratings in user u's long-term profile. For example, $\text{Bob}^{LT} = \{\text{Naked Gun}, \text{Aliens}\}$.

There are many ways of building collaborative recommenders. Here we describe just the one we have implemented; for details, see [2]:

- The similarity $w_{u_a,u}$ between the active user u_a and each other user, $u \neq u_a$, is computed using Pearson Correlation, $\text{correl}(u_a, u)$, over their co-rated items, devalued in the case of users who have co-rated fewer than 50 items by a significance weight. For later parts of this paper, it is important to appreciate that $\text{correl}(u_a, u)$ and hence $w_{u_a,u}$ can be positive, zero or negative.
- Next, the *nearest neighbours* of u_a are selected, i.e. the N (in our case, 20) other users u for whom $w_{u_a,u}$ is highest.
- For each item i that has not been rated by u_a but has been rated by at least one of the neighbours, u_a's rating for i is predicted, p_{i,u_a}, essentially as a weighted average of the neighbours' ratings for item i.
- These items are then sorted into descending order of p_{i,u_a}. This is the order in which items will be recommended. For example, if in a single-shot system we want to recommend three items, then the first three items in this sorted list are selected.

3 The RS-CCR$^+$ and RS-CCR$^\pm$ Systems

RS-CCR$^+$ and RS-CCR$^\pm$ are our designations of the very basic conversational collaborative recommender systems described in [5]. Note, however, that the system with which Rafter & Smyth do their experiments is closer to the one we describe in the next section (B.Smyth, personal communication 2005). In all these systems, the active user has a long-term profile (based on a column in the ratings matrix), u_a^{LT}, as do all other users. But, for the duration of her interaction with the system, the active user also has two short-term profiles, $u_a^{ST^+}$ and $u_a^{ST^-}$.

Initially, the short-term profiles are empty and the first set of k (typically, three) recommendations is made in the fashion described in Sect. 2. At this point, and indeed whenever the system recommends items to the user, the system solicits user feedback. The user's feedback takes one of two forms:

- She can indicate which recommended item best matches her short-term interests. If she does, the selected item's identifier is added to her short-term positive profile, $u_a^{ST^+}$. Nothing is done with the other items.
- She can indicate that none of the recommended items adequately meets her short-term interests. If she does, the identifiers of all the recommended items are added to her short-term negative profile, $u_a^{ST^-}$.

The system then recommends another set of items. New recommendations never repeat ones made previously in this dialogue. But, additionally, through the way it computes user similarity, the system attempts to steer new recommendations towards the kind of items in $u_a^{ST^+}$ and away from the kind of items in $u_a^{ST^-}$; see below for details. This recommendation-feedback cycle continues until either the user finds an item she wishes to consume, she abandons the dialogue having found no such item, or the system can make no fresh recommendations.

It remains to say how $u_a^{ST^+}$ and $u_a^{ST^-}$ influence subsequent recommendations. When finding neighbours, users whose long-term profiles contain items in the active user's short-term profiles will receive a boost:

- In RS-CCR$^+$, the more a user's long-term profile overlaps with the active user's short-term positive profile, the greater the boost:

$$w_{u_a,u} =_{\text{def}} \text{correl}(u_a, u) \times \text{overlap}(u_a^{ST^+}, u^{LT}) \tag{1}$$

- In RS-CCR$^\pm$, overlaps with the active user's short-term positive and negative profiles are combined by H, the harmonic mean:

$$w_{u_a,u} =_{\text{def}} \text{correl}(u_a, u) \times H(\text{overlap}(u_a^{ST^+}, u^{LT}), \text{overlap}(u_a^{ST^-}, u^{LT})) \tag{2}$$

Given that $u_a^{ST^+}$, $u_a^{ST^-}$ and u^{LT} are simply sets of item identifiers, the overlap function is defined as the size of the intersection of its two arguments (R.Rafter, personal communication 2004). If either intersection is empty, as they quite commonly will be, then in Equations (1) and (2) correl(u_a, u) will be multiplied by zero, making $w_{u_a,u}$ also zero. This is undesirable. Hence, Rafter & Smyth do not use the overlap when it is zero (B.Smyth, personal communication 2005) and, with the same effect, in our implementations of RS-CCR$^+$ and RS-CCR$^\pm$ we use the following:

$$\text{overlap}(A, B) =_{\text{def}} \max(1, |A \cap B|) \tag{3}$$

which defaults to 1 when the intersection is empty. (Other definitions are possible without making any major difference to the results, e.g. $1 + |A \cap B|$.)

4 The CCR$^+$ and CCR$^\pm$ Systems

The idea in conversational collaborative recommending is that the selection of nearest neighbours is "...directed towards users that have liked the items in the target user's [short-term positive profile], and towards users that have disliked items in the target user's [short-term negative profile]." [5, p.152] However, in the systems we have described in Sect. 3 the items in the active user's short-term positive profile $u_a^{ST^+}$ are compared with *all* the items in the other user's long-term profile u^{LT}, irrespective of whether the other user liked them or not. Similarly, items in $u_a^{ST^-}$ are compared with the whole of u^{LT}, rather than just those members of u^{LT} that u disliked.

To remove this weakness, the systems that we designate CCR$^+$ and CCR$^\pm$ partition u^{LT} into two: the likes and the dislikes. The likes (the long-term positive profile) we denote by u^{LT^+} and this set is compared with $u_a^{ST^+}$. The dislikes (the long-term negative profile) we denote by u^{LT^-} and this set is compared with $u_a^{ST^-}$. For example, in the MovieLens dataset, whose rating scale is 1–5, u^{LT^+} contains items rated 3 or above; u^{LT^-} contains items rated below 3. Before we give the new definitions of $w_{u_a,u}$, there is another issue to resolve.

Recall that $\text{correl}(u_a, u)$ can be negative. On occasion, there may be so few positively correlated neighbours that negatively correlated users are among those with the highest values for $\text{correl}(u_a, u)$. When this is the case, Equations (1) and (2) will multiply this negative number by the overlap, which is positive. Far from boosting the similarity of a user with high overlap, the resulting value for $w_{u_a,u}$ will be a negative number of greater magnitude and so the user will be less likely to be a neighbour. To obtain proper boosting behaviour, we have chosen to *add* overlap values to $\text{correl}(u_a, u)$. (We tried some better-motivated schemes, but they worked less well.)

Taking both the above ideas into account, we define $w_{u_a,u}$ as follows:

- In CCR$^+$:

$$w_{u_a,u} =_{\text{def}} \text{correl}(u_a, u) + \text{overlap}(u_a^{ST^+}, u^{LT^+}) \tag{4}$$

- In CCR$^\pm$:

$$w_{u_a,u} =_{\text{def}} \text{correl}(u_a, u) + \text{overlap}(u_a^{ST^+}, u^{LT^+}) + \text{overlap}(u_a^{ST^-}, u^{LT^-}) \tag{5}$$

Rafter & Smyth have also addressed the two issues we have discussed in this section (B.Smyth, personal communication 2005). Like us, they partition u^{LT} into likes and dislikes, comparing the former with $u_a^{ST^+}$ and the latter with $u_a^{ST^-}$. Their way of overcoming the problem of negative values of $\text{correl}(u_a, u)$ is to exclude such users u from the set of neighbours. We chose our approach because excluding negatively correlated users will (slightly) narrow the set of items that may be recommended and we felt that this was undesirable given that the goal of the system is to make recommendations that do not necessarily reflect the user's normal long-term interests. However, we suspect that this difference of detail results in only marginal differences in recommendations.

In any case, CCR^+ and $\text{CCR}^\pm$ perform only slightly better than RS-CCR^+ and RS-$\text{CCR}^\pm$(see Sect. 6). A more radical innovation is needed.

5 The CCR^+-Div and $\text{CCR}^\pm$-Div Systems

This section introduces the CCR^+-Div(b, k) and $\text{CCR}^\pm$-Div(b, k) systems. In their names, Div indicates a concern for the diversity of recommendations; b and k are parameters, which are explained below.

For content-based recommender systems, the argument has been convincingly made that items should be selected for *diversity* (relative to each other) as well as *similarity* (to the query or the user's profile) [7]. Too much similarity between the recommended items (e.g. three Woody Allen movies) can be undesirable. But, when recommendations are diverse, if the user is not satisfied with the most highly recommended item, for example, the chances of her being satisfied with one of the alternative recommendations is increased.

There is a body of research that addresses diversity for content-based recommenders, e.g. [7, 1, 4]. It is only now that we are seeing the first work that attempts to improve the diversity of the items recommended by collaborative recommenders. Specifically, apart from our own work, we are aware only of Ziegler's work on book recommendations [8]. Neglect of diversity may be because collaborative recommenders can provide *serendipitous* recommendations [2]. Serendipitous recommendations are pleasing recommendations for unexpected items; on occasion, they may increase diversity. However, we hypothesise that a more direct concern for diversity may be important, especially in *conversational* collaborative systems.

To investigate this, we implemented the Bounded Greedy selection algorithm (henceforth BG) from [7]. To recommend k items, BG finds bk items. In [7], these are the bk items that are most similar to the query (content-based recommending). Here, they are the bk items with the highest prediction values p_{i,u_a} (where neighbours are computed by the CCR^+ or $\text{CCR}^\pm$ systems). From these bk items, BG selects k to recommend to the user. It selects the k in a greedy fashion, based on ones selected so far; see Algorithm 1.

In the algorithm, the quality of item i relative to the result set so far R is defined as follows:

$$\mathit{Quality}(i, R) =_{\text{def}} (1 - \alpha) \times p_{i,u_a} + \alpha \times \mathit{RelDiv}(i, R) \tag{6}$$

Algorithm 1 The Bounded Greedy selection algorithm. Adapted from [7].

```
$Candidates \leftarrow bk$ items recommended by CCR$^+$ (or CCR$^\pm$)
$R \leftarrow \{\}$
for $j \leftarrow 1$ to $k$ do
    $best \leftarrow$ the $i \in Candidates$ for which $Quality(i, R)$ is highest
    insert $best$ into $R$
    remove $best$ from $Candidates$
end for
return $R$
```

α is a factor that allows the importance of the predicted rating and diversity to be changed; we use $\alpha = 0.5$. Diversity relative to the result set so far is defined as the average distance between i and the items already inserted into R:

$$\mathit{RelDiv}(i, R) =_{\text{def}} \begin{cases} 1 & \text{if } R = \{\} \\ \frac{\sum_{j \in R} \text{dist}(i,j)}{|R|} & \text{otherwise} \end{cases} \tag{7}$$

This leaves the issue of how to measure distance *between items* in Equation (7). In [7], the distance between items is the inverse of the *content-based* similarity. If item descriptions are available, the same approach can be used to enhance the diversity of collaborative recommendations. Ziegler, for example, uses taxonomic knowledge in his system [8]. But we choose to proceed on the assumption that item descriptions are not available. We enhance diversity using a measure of distance that is calculated using *collaborative data only*.

The intuition behind our approach to distance is that the community of users who have rated item i have a certain set of tastes. The more the membership of the community who rated item i differs from the membership of the community who rated item j, the more likely i and j satisfy different tastes and are different kinds of items.

In detail, then, we compute $\text{dist}(i, j)$ as follows:

- CCR$^+$(or CCR$^\pm$) will already have found u_a's N nearest neighbours.
- For both i and j, we create bit vectors I and J of length N. Digit d in vector I is set if neighbour d has a non-$\perp$ rating for item i; similarly for bits in J.
- $\text{dist}(i, j)$ is computed as the Hamming distance between I and J, i.e. a count of the number of positions in which their bits differ.

Fig. 1 illustrates this process; it shows Naked Gun to be more different from Cape Fear than Taxi Driver is. In the figure, we take N, the number of nearest neighbours, to be 3, and we assume these are Ann, Col and Deb. We take their ratings from Table 1 and set bits to show who rated what.

There are other ways of computing distances between items, even using just collaborative data. We mention two alternatives and defend our own choice:

- We could compute the (inverse of) Pearson correlation between *rows* in the ratings matrix, Table 1. Some collaborative recommenders work on this basis, e.g. [6]. An advantage would be that item-item (dis-)similarities could

	Ann	Col	Deb
Cape Fear	0	1	1
Naked Gun	1	0	1

↘ ↓ ↙

Hamming distance: 2

	Ann	Col	Deb
Cape Fear	0	1	1
Taxi Driver	0	1	1

↘ ↓ ↙

Hamming distance: 0

Fig. 1. Hamming distances.

then be computed in advance and cached, needing recalculation only when a new rating arrives. However, our approach restricts attention to the nearest neighbours (which are not known until recommendation time), making item distances sensitive to the active user's long- and short-term preferences.

- Even so, with attention restricted to the nearest neighbours, we could still have used (inverse) Pearson correlation, instead of Hamming distance. The former has the advantage of being sensitive to the actual ratings (the numeric values). However, the BG algorithm requires a very large number of distances to be computed.[1] Hamming distance proves effective (see the next section) while being cheaper to compute.

6 Empirical Evaluation

We adopt Rafter's & Smyth's methodology [5], but our datasets differ. They select the 2100 largest user profiles from the '1 Million MovieLens Dataset'; the average profile size for the 2100 users is 355 ratings. We use the entire '100K MovieLens Dataset', which contains profiles for 943 users; the average profile size is 106 ratings, which we think is more realistic.[2]

One hundred user profiles are selected at random and removed from the dataset. Each of these will act in turn as an (artificial) active user. The item that the user is seeking is obtained through the leave-one-out methodology, i.e. given the active user's long-term profile, each item in turn is withheld and treated as the target item. Sets of 3 recommendations are made to the user until either the target item is one of the recommended items, there have been 100 recommendation cycles, or no further recommendations can be made to this user, whichever comes soonest. If the target item is recommended within 100 cycles, the number of items recommended is recorded. Results are subjected to 3-fold cross-validation, with a different 100 active users in each fold.

In each recommendation cycle, the (artificial) user's feedback needs to be simulated. For each movie, the MovieLens datasets record a set of genres, which allows a simple-minded content-based approach. If the target item's set of genres is G_t and a recommended item's set of genres is G_r, we compute $\frac{|G_t \cap G_r|}{|G_t \cup G_r|}$. If all recommended items score zero, then none is taken to match the user's short-term

[1] $\frac{(2b(k-1)-k-1)k}{2}$ of them, in fact, for each set of k recommendations!

[2] We are grateful to the GroupLens project team for making their data available.

interests, so all the items are inserted into $u_a^{ST^-}$; otherwise, the highest-scoring item (with random tie-breaking) is taken to match the user's short term-interests, so this item is inserted into $u_a^{ST^+}$ and nothing is done with the others.

Fig. 2a shows, as a percentage of 34759 dialogues, how often the target item was found. In addition to RS-CCR$^+$, RS-CCR$^\pm$, CCR$^+$, CCR$^\pm$, CCR$^+$-Div(3, 15) and CCR$^\pm$-Div(3, 15), we show the results for SS-CR, a single-shot recommender (Sect. 2). We regard SS-CR as successful if the target item is among all the possible recommendations it can make to the active user. The other systems are successful if the target item is recommended within 100 cycles of 3 recommendations each. Unsurprisingly, SS-CR has by far the highest success rate; encouragingly, the diversity-enhanced systems, CCR$^+$-Div(3, 15) and CCR$^\pm$-Div(3, 15), have higher success rates than the others.

Fig. 2b shows, for each system's successful trials, how many items are recommended, on average, before the system recommends the target item. The diversity-enhanced systems recommend 20 fewer items than the best of the others. However, all the systems recommend, on average, over 100 items before they reach the target. This would clearly not be acceptable in practice. In defence, we note that the experimental methodology is severe: real users might be satisfied with any one of a set of items, whereas in the experiments there is a single target item each time. Furthermore, the simulated user feedback is so crude that it can sidetrack the conversational recommenders, making them on occasion uncompetitive even with the single-shot system.

Figs. 2c and 2d compare each system with SS-CR (when both are successful). We see (Fig 2c) that the diversity-enhanced systems make fewer recommendations than SS-CR nearly 80% of the time; the other systems are competitive with SS-CR less than 40% of the time. Then in Fig. 2d we show winning and losing margins. The figure shows, for example, that, when RS-CCR$^\pm$ wins against SS-CR, it makes on average 52 fewer recommendations and, when RS-CCR$^\pm$ loses against SS-CR, it makes on average 40 more recommendations. By this measure, RS-CCR$^\pm$ and RS-CCR$^+$ win by most when they win, but they also lose by most when they lose.

Fig. 2e compares the diversity-enhanced systems (with different values for b) with systems that choose k products *at random* from the bk products that have the highest predicted ratings (designated CCR$^+$-Rnd(b, k) and CCR$^\pm$-Rnd(b, k)). This allows us to see that our diversity-enhancement mechanism is making a systematic improvement. We also note that higher values of b have the greatest advantage.

Finally, in Fig. 2f, we compute for each system the average diversity (all-pairs distance) of each set of items it recommends, averaged over all such sets. CCR$^+$-Div(b, k) and CCR$^\pm$-Div(b, k) have the best values but, of course, this has to be taken with a pinch of salt, because it evaluates these two systems with exactly the measure that they seek algorithmically to maximise! All the values may seem low but this is a facet of the averaging; some of the individual recommendation sets may be quite diverse.

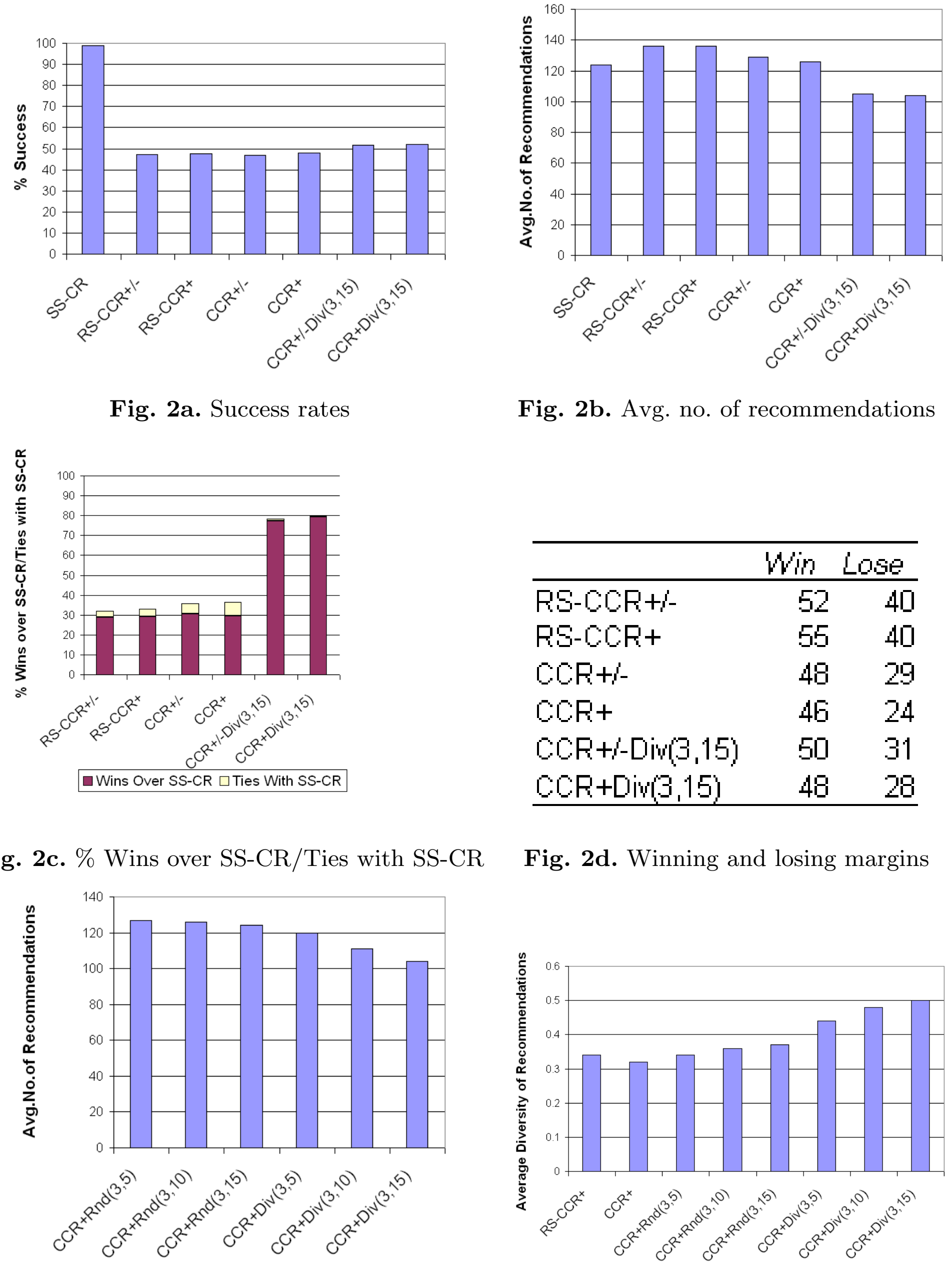

Fig. 2a. Success rates

Fig. 2b. Avg. no. of recommendations

	Win	*Lose*
RS-CCR+/-	52	40
RS-CCR+	55	40
CCR+/-	48	29
CCR+	46	24
CCR+/-Div(3,15)	50	31
CCR+Div(3,15)	48	28

Fig. 2c. % Wins over SS-CR/Ties with SS-CR

Fig. 2d. Winning and losing margins

Fig. 2e. Avg. no. of recommendations

Fig. 2f. Avg. diversity of recommendations

Fig. 2. Empirical results

7 Conclusions

Building on the seminal work reported in [5], we have developed a number of conversational collaborative recommender systems. In all these systems, the selection of neighbours is guided by overlap with the active user's short-term positive and negative profiles. In CCR$^+$-Div(b, k) and CCR$^\pm$-Div(b, k), we introduce for the first time an explicit mechanism that uses collaborative data only to enhance the diversity of recommendations made by (conversational) collaborative recommender systems. Experimental results are hugely improved, and we show that our diversity mechanism makes a systematic difference over a random selection.

Conversational collaborative recommenders are a new line of research, and enhancing the diversity of their recommendations is a new departure too. Future work could include: seeking better-motivated ways of boosting similarity; and more systematic investigation of good values for α, b and k; We would like to see an empirical comparison of different ways of computing item distance (some of which were mentioned in Sect. 5). It would be particularly interesting to compare approaches that use purely collaborative data with those that use content-based data. We would also like to investigate the role of diversity over the course of the dialogue. Diversity can be helpful in early cycles, when the user is exploring the space and making her short-term interests known; but in later cycles, when the user is homing in on a suitable item, diversity may be less appropriate [3].

References

1. D. Bridge and A. Ferguson. Diverse product recommendations using an expressive language for case retrieval. In S. Craw and A. Preece, editors, *Procs. of the 6th European Conference on Case-Based Reasoning*, pages 43–57. Springer, 2002.
2. J. L. Herlocker. *Understanding and Improving Automated Collaborative Filtering Systems.* PhD thesis, University of Minnesota, 2000.
3. L. McGinty and B. Smyth. On the role of diversity in conversational recommender systems. In K. Ashley and D. Bridge, editors, *Procs. of the 5th International Conference on Case-Based Reasoning*, pages 276–290. Springer, 2003.
4. D. McSherry. Diversity-conscious retrieval. In S. Craw and A. Preece, editors, *Procs. of the 6th European Conference on Case-Based Reasoning*, pages 219–233. Springer, 2002.
5. R. Rafter and B. Smyth. Towards conversational collaborative filtering. In L. McGinty and B. Crean, editors, *Procs. of the 15th Artificial Intelligence and Cognitive Science Conference*, pages 147–156, 2004.
6. B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl. Item-based collaborative filtering recommendation algorithms. In *Procs. of the Tenth International World Wide Web Conference*, pages 285–295, 2001.
7. B. Smyth and P. McClave. Similarity vs. diversity. In D. W. Aha and I. Watson, editors, *Procs. of the 4th International Conference on Case-Based Reasoning*, pages 347–361. Springer, 2001.
8. C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen. Improving recommendation lists through topic diversification. In *Procs. of the 14th International World Wide Web Conference*, 2005.