Interaction-Based Collaborative Filtering Methods for Recommendation in Online Dating

Alfred Krzywicki, Wayne Wobcke, Xiongcai Cai,
Ashesh Mahidadia, Michael Bain, Paul Compton and Yang Sok Kim

School of Computer Science and Engineering
University of New South Wales
Sydney NSW 2052, Australia
{alfredk,wobcke,xcai,ashesh,mike,compton,yskim}@cse.unsw.edu.au

Abstract. We consider the problem of developing a recommender system for suggesting suitable matches in an online dating web site. The main problem to be solved is that matches must be highly personalized. Moreover, in contrast to typical product recommender systems, it is unhelpful to recommend popular items: matches must be extremely specific to the tastes and interests of the user, but it is difficult to generate such matches because of the two way nature of the interactions (user initiated contacts may be rejected by the recipient). In this paper, we show that collaborative filtering based on interactions between users is a viable approach in this domain. We propose a number of new methods and metrics to measure and predict potential improvement in user interaction success, which may lead to increased user satisfaction with the dating site. We use these metrics to rigorously evaluate the proposed methods on historical data collected from a commercial online dating web site. The evaluation showed that, had users been able to follow the top 20 recommendations of our best method, their success rate would have improved by a factor of around 2.3.

1 Introduction

Online dating is a suitable application domain for recommender systems because it usually has abundant information about users and their behaviour. The key to developing a recommender system is that matches must be personalized. Moreover, in contrast to typical product recommender systems, it is unhelpful to recommend popular users; matches must be extremely specific to the tastes and interests of the user. This is particularly important because, unlike product recommender systems, the receivers of messages may reject invitations from the user, whereas products are typically commodity items that can be reproduced and sold any number of times.

The search and simple user profile matching technologies used in many dating sites have serious limitations. The main limitations are that some users receive large amounts of potentially unwanted communication (especially, perhaps, if they are the most popular users), whereas others may receive very low amounts of communication (e.g. if their profiles are not easily found via common search keywords). Search interfaces typically do not rank matches by suitability, requiring users to manually filter the results in order to find a potential partner. These aspects reduce the usability of the site, in different ways for different classes of users, and may lead to an increased attrition rate of users.

A recommender system can improve user satisfaction by addressing some of the limitations of search and user profile matching. By ranking potential matches, some of the burden faced by users in searching and filtering through candidates can be reduced, ideally increasing interactions and communication with suitable matches, plus increasing satisfaction for those less popular users who are not prominent on the site. In addition, by recommending such less popular users (where appropriate), the large volumes of unwanted communication may be reduced, though not eliminated, as such a recommender system would not replace, but augment, the existing search interfaces.

This work concerns the type of online dating site where users can freely and easily browse large numbers of user profiles. In broad terms, the typical "user experience" with such an online dating site evolves from initial setting up of a profile, to sending messages (usually selected from a predefined set) to prospective partners, who may accept or reject these initial approaches, then, for those accepted, proceeding to full e-mail communication. At each stage, there is the potential for users to receive recommendations and as the quantity of information increases, the quality of the matches should also increase. Helping users to progress to the later stages in this process can achieve the aims of increasing user retention, maintaining the pool of candidate matches and improving user satisfaction. In this paper, we focus on the stage in this process where users initiate contact with potential partners, since we have reliable data concerning these interactions, and suitable metrics for evaluating the success of recommendations.

Content-based recommender systems make recommendations for a user based on a profile built up by analysing the content of items that users have chosen in the past. These content-based methods stem from information retrieval and machine learning research and employ many standard techniques for extracting and comparing user profiles and item content [6]. Initial experiments showed that it is difficult to use the user profiles for content-based recommendation because users do not always provide adequate information about themselves or provide such information in the form of free text, which is hard to analyse reliably. Another well-known significant limitation of content-based recommender systems is that they have no inherent method for generating recommendations of novel items of interest to the user, because only items matching the user's past preferences are ever recommended [8, 2, 1].

Collaborative filtering recommender systems address some of the problems with content-based approaches by utilizing aggregated data about customers' habits or preferences, recommending items to users based on the tastes of other "similar" users [8]. In this paper we show that collaborative filtering based on interactions between users is a viable approach to recommendation in online dating. This result is surprising because our collaborative filtering methods rely only on interactions between users and do not explicitly incorporate user profile information. We propose a number of new collaborative filtering methods and metrics specifically designed to measure user success on an online dating site, and show that a "voting" method used to rank candidates works efficiently and generates reasonably good recommendations. This paper is organized as follows. In the next section we provide a brief review of related research in this domain. Then we describe new collaborative filtering methods applicable in the online dating domain. In Section 4, we define a set of metrics and evaluate and compare all methods, then conclude with an outline of future research.

2 Related Work

We are not aware of any published work that uses pure collaboration filtering in the online dating domain. By pure collaboration filtering we mean user-to-user or item-to-item methods that are based only on user interactions. Brǒzovský and Petřıček [3] evaluate a number of collaborative filtering methods in an online dating service for estimating the "attractiveness" rating of a user for a given user, which is calculated from the ratings of similar users. Their results, however, are not directly comparable with our research since we focus on the problem of predicting *successful* interactions, which requires taking into account the interests of the receiver of a message in addition to the preferences of the sender, which is a two way interaction, in contrast to "attractiveness" ratings which only reflect the viewer's interests. In particular, whereas many people are likely to rate the same people highly, recommending these same people to many users is likely to result in a large number of unsuccessful interactions, since typically any person is limited in the number of successful responses they can give.

Generally, collaborative filtering methods differ in how the recommendations are derived and how they are ranked. The item-to-item collaborative filtering method of Amazon.com [5] uses a similarity table computed offline and the ranking is based on the similarity between items, which is related to the number of customers who purchased the two items together. For all of our methods, we compute both a similarity table and a recommendation table offline and rank the candidates not according to their similarity, but based on number of unique "votes" as described in the next section. Other methods of calculating similarity used in the literature include Pearson correlation coefficient and recency [9], cosine and conditional probability [4], and cosine, correlation and sum of user ratings weighted by similarity [7].

3 Methods

3.1 Basic Collaborative Filtering

The *Basic Collaborative Filtering* (Basic CF) item-to-item model is based on similar items: two items are similar if they have been purchased together in a shopping basket by a large number of customers, and (simplifying) the number of such purchases indicates the degree of similarity between the two items. By analogy, in the context of initiating contact in an online dating site there are similar senders and similar recipients, i.e. instead of purchasing items, the basic interaction is that users send messages to other users which may be successful (accepted) or unsuccessful (rejected). Two distinct users are *similar recipients* to the extent that they have received messages from the same senders; two distinct users are *similar senders* to the extent that they have sent messages to the same recipients. More formally, let $u \rightarrow v$ denote an interaction where u sends a message to v. The similar recipients $u \overset{r}{\sim} v$ and similar senders $u \overset{s}{\sim} v$ relations can be defined as follows:

$$u \overset{r}{\sim} v : \exists w\, (w \rightarrow u \wedge w \rightarrow v) \tag{1}$$

$$u \overset{s}{\sim} v : \exists w\, (u \rightarrow w \wedge v \rightarrow w) \tag{2}$$

Both types of similarity can be used to make recommendations, as shown in Figure 1. In these diagrams, rectangles (blue background) typically indicate senders, ellipses (pink background) indicate recipients. Unbroken (black) arrows indicate messages sent between users, and (green) lines indicate the similarity of users (as recipients and senders, respectively).

In the first diagram in Figure 1, all pairs of recipients except r1 and r4 are similar recipients, since for any pair, both users in the pair have received at least one message from the same sender; the number of such messages indicates the degree of similarity as recipients. So for example, r2 is a similar recipient to r3 to degree 2 since both have received contacts from s1 and s2. The same reasoning implies that s1 and s2 are similar senders; s1 and s2 are similar senders to degree 2 since they have both sent messages to r2 and r3.

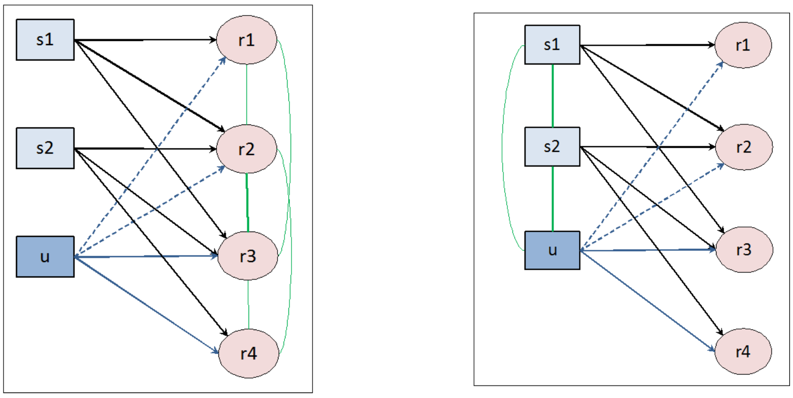

Fig. 1. Basic CF Recommendation with Similar Recipients and Similar Senders

Now consider another user u who has already sent messages to r3 and r4. Basic CF item-to-item recommendation says that recipients similar to either r3 or r4 should be recommended to u. In general, the method generates a set of candidate pairs $\langle u, c\rangle$, giving a set of candidates c for each of a set of users u. If $u \nrightarrow v$ means that u has not contacted v, the set of candidate pairs can be defined by the following expression:

$$C = \{\langle u, r\rangle : \exists r' \, (u \rightarrow r' \wedge r' \overset{r}{\sim} r \wedge u \nrightarrow r \wedge r \nrightarrow u)\} \tag{3}$$

where u is the user to whom we recommend another user r.

These recommendations can be ranked by various means, however, we choose to rank them in the following way: each candidate recipient r who is a similar recipient to either r3 or r4 has a number of "votes" being the number of recipients (out of r3 and r4) to which r is a similar recipient; the ranking of r is just the number of these votes. In the first diagram in Figure 1, since r1 and r2 are similar to either r3 or r4, both can be recommended to u; r1 has 1 vote since r1 is similar to r3 but not r4, and r2 has 2 votes since r2 is similar to both r3 and r4. Where $|\cdot|$ denotes set cardinality, the number of votes of a candidate r for a user u is defined as follows:

$$votes(u, r) = |\{r' : (r \overset{r}{\sim} r' \wedge u \rightarrow r')\}| \tag{4}$$

Recommendation using similar senders yields exactly the same set of candidates, but with a different ranking, where this ranking is determined by the votes of similar senders.

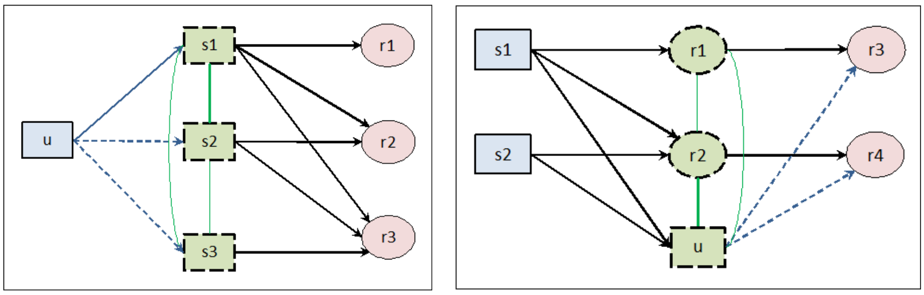

Fig. 2. Inverted CF Recommendation with Similar Senders and Similar Recipients

By symmetry, there are two other collaborative filtering recommendation methods, which we call "inverted" methods, for both similar senders and similar recipients. In Figure 2, thick broken line bordered shapes (green background) represent users who are both senders and recipients (for the purposes of the discussion, though of course senders also receive messages from other users and vice versa). In the first diagram in Figure 2, as in Basic CF recommendation using similar recipients, users are recommended to u if they are similar to u's contacts; however, in contrast to the standard approach, similarity here is based on similarity *as senders*. In the example, u has contacted s1, so s2 and s3 can be recommended to u, since they are both similar as senders to s1. The ranking is the number of u's contacts to whom they are similar (as senders), in this case both 1. One intuition underlying this inverted scheme is that it makes sense to recommend to u people with similar preferences (as determined by their sending patterns) to the contacts of u, as ideally, if those preferences indicate an interest in u, then those recommended users should also be highly inclined to reply positively to an invitation from u. Thus this method is a very strong filter on candidates compared to the other methods, in using the preferences of recipients as the basis of the recommendations. More formally, the candidate set and the vote calculation for this *Inverted CF Sender* model is as follows:

$$C = \{\langle u, s\rangle : \exists s' (u \rightarrow s' \wedge s' \overset{s}{\sim} s \wedge u \nrightarrow s \wedge s \nrightarrow u)\} \tag{5}$$

$$votes(u, s) = |\{s' : (s' \overset{s}{\sim} s \wedge u \rightarrow s')\}| \tag{6}$$

Another inverted method is illustrated in the second diagram in Figure 2. As in the Inverted Sender method, for a user u, users are recommended who have received contacts from similar users to u, but this time the similarity to u is based on similarity *as recipients*. In the example both r3 and r4 can be recommended to u as they have received contacts from users similar to u. The rank of a candidate r is the number of users who are similar as recipients to u and who have contacted r, in this case 1 for both r3 and r4. An intuition here is that u and the similar users are perceived as similar by prospective senders, hence (again inverting the Basic CF scheme) are likely to receive positive replies from similar people (for example, such senders). The candidate set and the vote calculation for this *Inverted CF Recipient* model is shown below:

$$C = \{\langle u, r\rangle : \exists r' \, (r' \overset{r}{\sim} u \wedge r' \rightarrow r \wedge u \nrightarrow r \wedge r \nrightarrow u)\} \tag{7}$$

$$votes(u, r) = |\{r' : (r' \overset{r}{\sim} u \wedge r' \rightarrow r)\}| \tag{8}$$

An interesting point about this method is that users do not have to have sent any messages in order to receive recommendations; it is sufficient that they are similar as recipients to other users. Also note that for the inverted methods u and r may be both senders and recipients.

3.2 Extended Methods

The methods described in the previous section take all interactions (both successful and unsuccessful) into consideration in the calculation of "similarity" and in the ranking of candidates. In this section, we discuss variants of the basic methods that treat such positive and negative interactions separately for constructing the sets of candidates. We also consider combining candidate sets from different basic models. By a *positive* or *successful* interaction, we mean an interaction where user u sent a message to user v and received a reply message (again drawn from a predefined set) that was considered to encourage further communication. Other interactions are *negative* or *unsuccessful*.

We now consider variants of the basic and inverted collaborative filtering methods, denoted *Basic CF+*, *Inverted CF+ Sender* and *Inverted CF+ Recipient*, that consider only positive interactions for the generation of similarity and candidate sets. In addition, *Combined CF+* unifies the above three methods by adding the votes for candidate pairs that occur more than once. We also tried other methods for combining votes, such as taking the maximum and average of the votes of the methods, but found that simple addition gives the best results. Finally, *Best Two CF+* combines only Basic CF+ and Inverted CF+ Recipient in one candidate set, adding the votes for candidates generated by both methods. Another method called *Two Way CF+* is based on combining the two similarity relations by taking their intersection, so that pairs are similar as both senders and recipients. Recommendations for a user are generated from both senders similar to the user and recipients similar to the user's contacts. The idea was that this would create stronger candidates for recommendation. The formal definitions are as follows, where $u \overset{+}{\rightarrow} v$ means a positive contact from u to v.

$$u \overset{rs}{\sim} v \overset{\text{def}}{=} u \overset{r}{\sim} v \wedge u \overset{s}{\sim} v \tag{9}$$

$$C = C_r \cup C_s \tag{10}$$

where

$$C_r = \{\langle u, r\rangle : \exists r' \, (u \overset{+}{\rightarrow} r' \wedge r' \overset{rs}{\sim} r \wedge u \nrightarrow r \wedge r \nrightarrow u)\} \tag{11}$$

and

$$C_s = \{\langle u, r\rangle : \exists s\,(s \xrightarrow{+} r \wedge s \overset{rs}{\sim} u \wedge u \nrightarrow r \wedge r \nrightarrow u)\} \tag{12}$$

We also tested methods that use both positive and negative interactions. These methods construct two candidate sets: one based on positive votes using the Best Two CF+ method, and one using negative votes, also using the Best Two CF+ method but created using negative interactions. The two sets are then combined by subtracting the negative votes from the positive votes for a given candidate pair. Intuitively, the negative votes for a candidate to be recommended to a user count against the recommendation, in that the candidate has previously rejected other users who are similar to that user. We considered two ways of weighting the negative votes. In *Best Two CF+/-*, negative votes are weighted equally to positive votes. However, since there are many more negative interactions than positive interactions, this weighting is high. So we also considered a method *Best Two CF+/-0.2* in which negative votes were weighted 0.2 (there are roughly 5 times as many negative interactions as positive interactions, so the average number of votes for a moderate candidate is around 0).

4 Evaluation

In this section, we focus on evaluating the collaborative filtering methods described in Section 3 on historical data collected from a commercial dating web site. This dating site records user profiles and the interactions between users. User profiles contain basic information about users, such as age, location, education and occupation, information about user interests in the form of free text and information about a user's previous searches. The interaction information contains details of messages sent between users.

Note that users of the site did not have access to our recommendations, so the results are with respect to the users' actual behaviour on the site, with a view to comparing the various methods using several metrics, discussed below.

4.1 Evaluation Metrics

Any interaction instance that actually occurs has a correct classification (known from the historical data) as either successful or unsuccessful. However, in recommender systems, we are interested in the generation of sufficiently many *new* candidates with sufficiently high confidence of them being successful. In particular, in evaluating different methods, we are interested in three questions: (i) how well the method conforms to the reality of user interactions, (ii) how many recommendations can be made and with what distribution over the set of users, and (iii) how well the candidates are ranked. Note that since only historical data is used, it is not our objective to predict the interactions that actually occurred in the data (since our eventual aim is to recommend new candidates to users), but rather to use suitable metrics to compare different methods.

More formally, each method considers a set of users P in some training set, and generates a set of potential successful contacts between members of P, typically many more contacts than occurred. Let C be the set of candidate pairs generated by a given method. This implicitly gives a set of senders S and two sets of recipients, R (those potential recipients of contacts from S) and Q (those recipients who actually received contacts from S in a test period), where a contact from S means a contact from any member of S. Let $m(C)$ be the set of contacts in C that actually occurred in a given test set, $nm(C)$ be the size of $m(C)$, and $nm(C, +)$ be the number of contacts in $m(C)$ that were successful. Similarly, $n(S)$ is the size of S, $ns(S, R)$ is the number of contacts from a member of S to a member of R and $ns(S, R, +)$ is the number of successful contacts from S to R.

The first question is how well a method conforms to the user interactions occurring in the test period. Borrowing from information retrieval, the metrics are:

– *Precision:* The proportion of those predicted successful contacts C that occurred which were successful, i.e.

$$Precision = nm(C, +)/nm(C) \tag{13}$$

– *Recall:* The proportion of successful contacts sent by S that are predicted by the method, i.e.

$$Recall = nm(C, +)/ns(S, Q, +) \tag{14}$$

A related question is how many users can receive recommendations using the various collaborative filtering methods. This is a kind of coverage measure, but oriented towards users rather than contacts. This is measured with respect to the set of users who sent or received at least one message in the training period (i.e. those users to whom candidates could potentially be recommended) rather than with respect to the users in the test set. Since there is a high overlap between the senders and receivers during the training period (around half the receivers are also senders), there are around 128,000 such users. Let M be this set of users and let N be the subset of M of users who receive recommendations using a collaborative filtering method. The metric is:

– *Coverage:* The proportion of users from M who receive recommendations, i.e.

$$Cov = n(N)/n(M) \tag{15}$$

The second question is how many potential interactions are generated by a given method (and in particular, how many candidates are recommended for each user). For this we simply count the number of candidates generated for the different users, and plot the distribution of these counts.

The third question is how well the candidates are ranked by the "voting" schemes used with the collaborative filtering methods. The metric we use here is Success Rate Improvement, defined as follows.

– *Baseline Success Rate*: The proportion of contacts from senders S to all recipients Q that were successful, i.e.

$$SR(S, Q) = ns(S, Q, +)/ns(S, Q) \tag{16}$$

– *Success Rate*: The precision, the proportion of those predicted successful contacts C that occurred which were successful, i.e.

$$SR(C) = nm(C, +)/nm(C) \tag{17}$$

– *Success Rate Improvement (SRI)*: Success Rate/Baseline Success Rate, i.e.

$$SRI = SR(C)/SR(S, Q) \tag{18}$$

The success rate improvement is designed specifically to measure how a particular method improves the chances of successful interactions between users as determined on the test set. To evaluate the ranking of candidates, the set of recommended candidates for each user is limited to various bounds, 20, 40, 60, 80 and 100, and SRI is calculated over only those candidates. That is, the top N success rate is calculated using only at most N candidates per user and not the set of all candidate pairs in C.

4.2 Evaluation Methodology

All collaborative filtering methods were evaluated by constructing a candidate set using some initial training set and then evaluating the candidates on a smaller test set, disjoint from the training set. The aim is to measure the effectiveness of the different recommendation methods with reference to the successful interactions that occurred.

The training set we used consisted of about 1.3 million interactions covering 4 weeks. The test set consisted of about 300,000 interactions over a period of six days immediately following the training set period. We also did preliminary evaluation using training periods of 3, 2 and 1 week before the test period, but found the 4 week period optimal in terms of precision and coverage.

A major requirement is that candidate generation be reasonably efficient. There is clearly a trade-off between the period of prior contact data used in the methods to calculate similarity, and the number of users who can be recommended and for whom recommendations can be provided. This trade-off is amplified due to the large numbers of interactions for a small minority of "popular" users. If all users are considered, a very large number of candidates will be the already popular users and will be provided for the already popular users, since they are similar (as both senders and recipients) to very many users. However, this is counter to the purpose of a recommender system, since these are the users least likely to want a recommendation or to be recommended to anyone else, as they are already the users with the highest number of communications. In the evaluations reported here, we therefore remove all popular users as both candidates for recommendation and for receiving recommendations, where *popular* means "has received more than 50 messages in the last 30 days".

The training dataset was prepared by selecting all contact events created in a four week period, selected as the training period, and excluding those made by such popular users or sent to such popular users (or both). The test dataset consisted of contact events occurring in the six day period following the training period, called the test period, for which both sender and recipient were active during the training period (i.e. ignoring interactions between users who started sending or receiving messages in the test period). This leaves around 700,000 interactions in the training set and around 120,000 in the test set. There were around 60,000 senders and 110,000 recipients in the training set, and 25,000 senders and 47,000 recipients in the test set (there is a high overlap between senders and recipients).

4.3 Results for Different Methods

Table 1 shows some basic statistics for the similarity tables computed using the different definitions of similarity, after removing popular users. Shown are the number of similar pairs of users and the average and maximum number of similar users per user, followed by the number of users with only 1 similar, 2 similar, 3 similar and between 3 and 50 users similar. Recipient+ and Sender+ refer to similarity as recipients and senders but taking into account only positive interactions, and R+ ∩ S+ is their intersection, used in the Two Way CF+ method. Since there are many more recipients than senders (a ratio of around 2 to 1), and many more contacts than successful contacts (a ratio of around 6 to 1), the relativities are not surprising. However, the large absolute numbers indicate that there is a high overlap in sending patterns (intuitively, senders contacting the same groups of receivers). These numbers suggest that the main collaborative filtering methods can generate large numbers of candidates, since even users with only one other similar user can potentially receive a recommendation.

Table 1. Similarity of Users Under Various Schemes

	Recipient	Sender	Recipient+	Sender+	R+ ∩ S+
Similar pairs	57,859,748	7,789,252	2,240,656	616,568	9,680
Unique users	114,044	61,024	50,939	35,808	4,916
Average similar users	507	127	49	18	2
Maximum similar users	8,043	7,789	1,004	740	29
Users with 1 similar	957	980	3,059	3,218	2,931
Users with 2 similar	1,046	1,032	2,753	2,837	992
Users with 3 similar	1,081	971	2,550	2,513	404
Users with 3–50 similar	32,426	27,748	32,854	27,347	993

Table 2 shows a summary of results for the candidate generation methods if all generated candidates for each user are recommended (the “All” column) and if the top 20 are recommended (the “Top 20” column). Note that the candidates do not include those possible candidates whom the user has already contacted in the training period. Unsurprisingly, the Basic CF method generates the most candidates, while the Inverted CF methods generate perhaps more than expected. What is surprising is the recall of all the methods, which indicate that the contacting patterns of the users largely conform to the underlying assumptions of collaborative filtering (i.e. similar users really are contacting similar recipients). As expected, the methods based on positive only interactions generate fewer candidates, but with higher precision and SRI than the methods based on all contacts. Note that the Basic CF and Inverted CF Sender methods (and the corresponding methods based on positive contacts) can only generate recommendations for users who sent messages during the training period; the test set of users includes, in addition, a substantial number of users who, whilst not sending any messages during the training period, sent messages in the test period. Their messages can never count towards the recall for these methods. In contrast, the Inverted CF Recipient methods do generate candidates for a large number of such users.

Table 2. Summary of Results for Collaborative Filtering Methods

		All				Top 20		
Method	Baseline SR	SR	Recall	Coverage	SRI	SR	Recall	SRI
Basic CF	0.161	0.174	0.565	44.6%	1.08	0.184	0.0236	1.14
Inverted CF Recipient	0.179	0.22	0.269	82.6%	1.23	0.2	0.0098	1.12
Inverted CF Sender	0.161	0.216	0.188	38.6%	1.35	0.274	0.0098	1.7
Basic CF+	0.165	0.271	0.174	25.8%	1.64	0.359	0.016	2.18
Inverted CF+ Recipient	0.189	0.378	0.036	35.1%	2.00	0.45	0.0105	2.38
Inverted CF+ Sender	0.168	0.309	0.024	19.9%	1.83	0.308	0.0064	1.84
All Combined CF+	0.173	0.346	0.071	47.1%	2.00	0.399	0.0134	2.31
Best Two CF+	0.169	0.282	0.179	46.6%	1.67	0.394	0.0202	2.34
Best Two CF+/-	0.169	0.282	0.179	46.6%	1.67	0.386	0.0162	2.29
Best Two CF+/-0.2	0.169	0.282	0.179	46.6%	1.67	0.399	0.019	2.36
Two Way CF+	0.157	0.247	0.027	18.3%	1.58	0.273	0.0044	1.75

Considering only individual basic and inverted collaborative filtering methods, it is clear that methods based on positive interactions (Basic CF+, Inverted CF+ Recipient and Inverted CF+ Sender) outperform those based on all contacts. Of these three methods, the method of choice is Basic CF+, which has a high SRI and a reasonable recall for those users for whom it provides recommendations (much higher, at 57%, when considering only the senders in training set). Since this has an SRI of 1.64, if those users followed the recommendations of Basic CF+, their success rate would improve by 64%. This degree of SRI is somewhat surprising, since it implies that users, once they have replied positively to some contacts, will continue to do so for contacts initiated by similar senders. The results for Inverted CF+ Recipient are more surprising, though the recall is much lower than for Basic CF+.

Since Inverted CF+ Recipient generates candidates for users who have sent no messages (but have replied positively to some messages), combinations of the above methods were analysed. The next two lines in Table 2 show the results for all three methods based on positive contacts combined and for the best two combined (Basic CF+ and Inverted CF+ Recipient). It can be seen that Basic CF+ and Inverted CF+ Recipient complement each other very well, with the coverage increasing significantly to around 47%, maintaining the precision of Basic CF+ with an increase in recall. However, the combination of all three methods performs similarly to the best two combined.

The next two lines are for methods that combine positive and negative votes to give different rankings, and are discussed further below. The results for Two Way CF+ (with ranking determined by adding the votes of senders similar to the user and recipients similar to the user's contacts) are shown in the last row of the table. Despite the fact that this method is based on a stronger notion of similarity, the SRI and recall are lower than for the methods based on one way similarity. This is probably because the sparsity of the data makes two way similarity too strict a criterion for generating candidates, but also there is an asymmetry between senders and recipients, in that some users will respond favourably to contacts received but will not initiate contacts. Two Way CF+ therefore fails to generate many potential successful interactions found by the other methods.

The second question of interest is the distribution of the counts of candidates generated for the users. We show here the results for the Basic CF+ method (perhaps the best single method, not considering the combined methods). The distribution of counts is shown in Figure 3 for bands of 1000. The distribution is as expected, with a small number of users having a large number of candidates, and the number of candidates dropping rapidly for a larger number of users. For example, around 35,000 users receive 1000 or fewer recommendations, and most users receive 2000 or fewer recommendations.

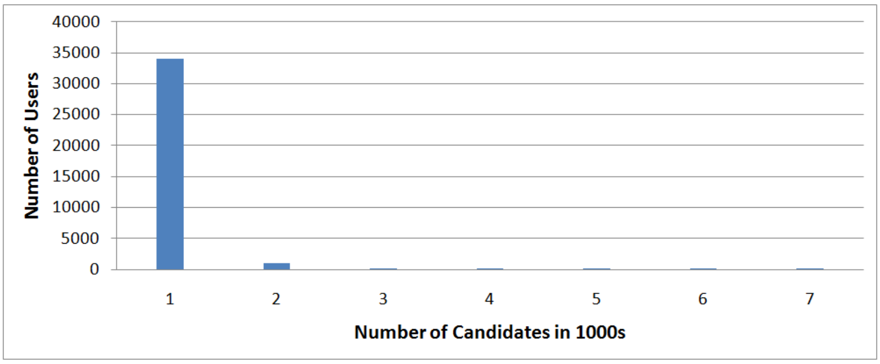

Fig. 3. Distribution of Candidate Counts for Basic CF+

Figure 4 shows the distribution of candidate counts for bands of 10 up to 100. Over 4000 users receive 10 or fewer recommendations; 3000 between 10 and 20, etc. However, this still means that over half of the approximately 60,000 senders from the training set receive more than 100 recommendations, and around 90% of senders from the training set receive more than 20 recommendations. These results show that Basic CF+ is capable of generating large numbers of candidates, and therefore the same applies to the combined methods, which generate even more candidates.

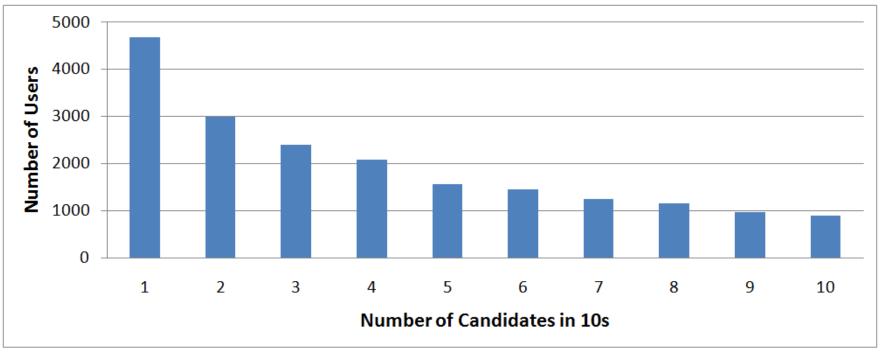

Fig. 4. Distribution of Candidate Counts within the 1-100 Range for Basic CF+

The final question is the ability of the "voting" methods used with collaborative filtering to rank candidates for recommendation. The question is whether the higher ranked candidates, being "closer" to the user's successful contacts in terms of interaction patterns, give a higher SRI than the recommended candidates in general. To answer this question, we calculated the SRI considering only the top N ranked candidates for each user. If the ranking is effective, the SRI should be higher for smaller values of N; while for large values of N, this SRI should be similar to the SRI when all candidates are considered. The baseline SRI of 1 represents the user's actual behaviour.

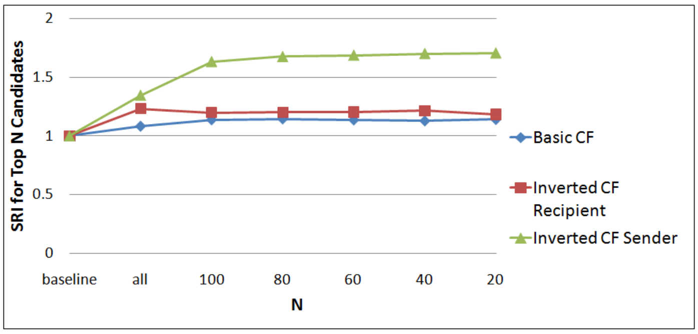

Fig. 5. SRI for Methods Based on All Contacts

Figure 5 shows the SRI for the top N candidates for various values of N up to 100 for methods based on all contacts (for Basic CF, the ranking is from the votes of similar recipients): 100 is the likely maximum number of candidates presented to any user, 20 is the number of candidates that fit in a single e-mail message. Inverted CF Sender provides some improvement in SRI, probably because recipient interest in the senders is taken into account, however this method has a low coverage. The ranking functions for the other CF methods show little improvement over the baseline. This is because these methods recommend many candidates who replied negatively to similar users and so also reply negatively to the user receiving the recommendation.

Figure 6 shows the SRI for the top N candidates for the methods based on positive contacts (with the ranking for Basic CF+ derived from the votes of similar recipients). As expected, the corresponding SRI values are all higher than in Figure 5. Surprisingly, Basic CF+ gives a high SRI with a reasonable coverage and the ranking function provides even more improvement over the baseline. More surprisingly, Inverted CF+ Recipient gives a high SRI with the ranking providing additional improvement, and the coverage is reasonable. The combination of the best two methods (Basic CF+ and Inverted CF+ Recipient), shown in Figure 7 as Best Two CF+, gives even better results (with the ranking calculated by adding the votes from the two methods). For the top 20 candidates of Best Two CF+, the SRI is around 2.3, meaning that if users had followed these recommendations, their success rate would improve from around 17% to nearly 40%. Moreover, as observed above, the coverage of users is quite high, around 50% of all senders and receivers in the training set.

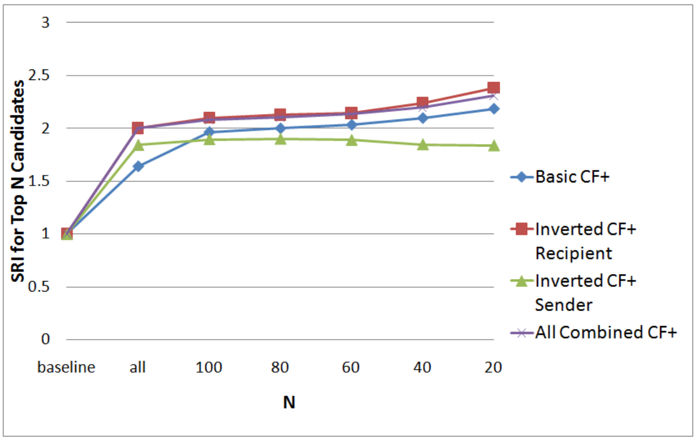

Fig. 6. SRI for Methods Based on Positive Contacts

The combination of all three methods (Figure 6), again adding the votes together to give the ranking, whilst having a slightly higher coverage, has a consistently lower SRI than the best two methods combined. As shown in Figure 7, Best Two CF+/- and Best Two CF+/-0.2 have similar SRI to Best Two CF+ but the recall for the top 20 is lower than that for Best Two CF+. Hence using negative votes does not improve the resulting recommendations, indicating that the fact that a similar user has been rejected by a candidate does not affect whether the user will be rejected by the candidate. The reduction in recall (Table 2) is more pronounced when the negative votes are weighted the same as positive votes compared to when they are weighted 0.2 (where there is probably very little difference in the final ranking of candidates). Thus the method combining Basic CF+ and Inverted CF+ Recipient is the clear method of choice as the basis of a recommender system.

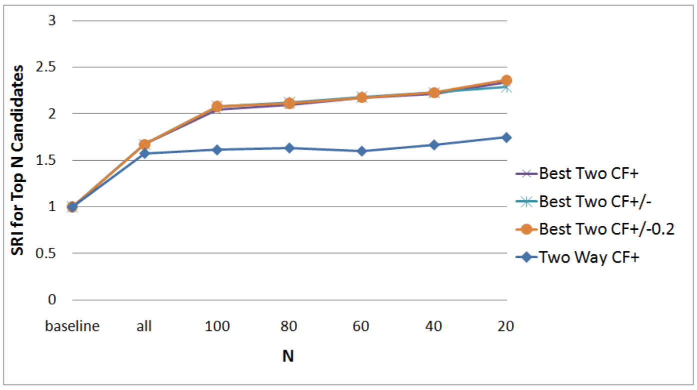

Fig. 7. SRI for Combined Methods

5 Conclusion and Future Work

We developed and evaluated a number of collaborative filtering methods designed for recommendation in online dating sites. We applied these methods to historical data from a dating web site and proposed a number of metrics to determine the effectiveness of our recommendation methods. Using these metrics we showed that collaborative filtering methods work very well when evaluated against the true (historical) behaviour of the users, not only for the generation of suitable candidates, but also to provide a substantial improvement in success rate over the baseline. This result is interesting because our collaborative filtering methods work only with user interactions and do not incorporate profile information. Our results imply that our methods (singly or combined) can be used as the basis of a recommender system for an online dating service. In particular, we found that combining the best two methods based on positive interactions provides an acceptable trade-off between success rate improvement, recall and coverage of users.

Future work includes applying machine learning methods to improve the ranking of recommended candidates, and hybrids of collaborative filtering and content-based methods to generate recommendations for users who do not have any successful interactions.

Acknowledgements

This work was funded by Smart Services Cooperative Research Centre. We would also like to thank our industry partners for providing the datasets.

References

1. Adomavicius, G., Kwon, Y.: New Recommendation Techniques for Multicriteria Rating Systems. IEEE Intelligent Systems 22(3), 48–55 (2007)
2. Balabanović, M., Shoham, Y.: Fab: Content-Based, Collaborative Recommendation. Communications of the ACM 40(3), 66–72 (1997)
3. Brŏzovský, L., Petřiček, V.: Recommender System for Online Dating Service. In: Proceedings of Znalosti 2007 (2007)
4. Karypis, G.: Evaluation of Item-Based *Top-N* Recommendation Algorithms. In: Proceedings of the Tenth International Conference on Information and Knowledge Management, pp. 247–254 (2001)
5. Linden, G., Smith, B., York, J.: Amazon.com Recommendations: Item-to-Item Collaborative Filtering. IEEE Internet Computing 7(1), 76–80 (2003)
6. Pazzani, M., Billsus, D.: Content-Based Recommendation Systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) The Adaptive Web. Springer-Verlag, Berlin (2007)
7. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-Based Collaborative Filtering Recommendation Algorithms. In: Proceedings of the 10th International World Wide Web Conference, pp. 285–295 (2001)
8. Shardanand, U., Maes, P.: Social Information Filtering: Algorithms for Automating "Word of Mouth". In: Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'95), pp. 210–217 (1995)
9. Zhou, D.X., Resnick, P.: Assessment of Conversation Co-mentions as a Resource for Software Module Recommendation. In: Proceedings of the 2009 ACM Conference on Recommender Systems, pp. 133–140 (2009)