Collaborative Filtering for People to People Recommendation in Social Networks

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

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

Abstract. Predicting people other people may like has recently become an important task in many online social networks. Traditional collaborative filtering approaches are popular in recommender systems to effectively predict user preferences for items. However, in online social networks people have a dual role as both "users" and "items", e.g., both initiating and receiving contacts. Here the assumption of active users and passive items in traditional collaborative filtering is inapplicable. In this paper we propose a model that fully captures the bilateral role of user interactions within a social network and formulate collaborative filtering methods to enable people to people recommendation. In this model users can be similar to other users in two ways – either having similar "taste" for the users they contact, or having similar "attractiveness" for the users who contact them. We develop SocialCollab, a novel neighbour-based collaborative filtering algorithm to predict, for a given user, other users they may like to contact, based on user similarity in terms of both attractiveness and taste. In social networks this goes beyond traditional, merely taste-based, collaborative filtering for item selection. Evaluation of the proposed recommender system on datasets from a commercial online social network show improvements over traditional collaborative filtering.

Keywords: Machine Learning, Recommender Systems, Collaborative Filtering.

1 Introduction

Traditional social filtering or *recommender* systems attempt to discover user preferences over items by modelling the relation between users and items. The aim is to recommend items that match the *taste* (likes or dislikes) of users in order to assist the active user, i.e., the user who will receive recommendations, to select items from an overwhelming set of choices. Such systems have many uses in e-commerce, subscription based services and other online applications, where provision of personalised suggestions is required [8]. By applying recommendation techniques it is possible to greatly increase the likelihood of the successful purchase of products or services by the active user, since services or products are

J. Li (Ed.): AI 2010, LNAI 6464, pp. 476–485, 2010.

personalised and presented to the active user using information obtained from the purchasing behaviour of like-minded users. In online applications with a very large number of choices where customer taste is important in making selections, personalised recommendation of items or people becomes essential.

1.1 Recommender Systems

Approaches to recommender systems can be categorised as content-based or collaborative filtering methods. In content-based methods, the user will be recommended items similar to those the user preferred in the past. This is usually based on models created from item descriptions using information retrieval or machine learning techniques. In general, a content-based system analyses the content of the profiles, or descriptions, of items, as well as provided user ratings, to infer a model that can be used to recommend additional items of interest. In this paper we do not address content-based recommendation.

Collaborative filtering (CF) methods, on the other hand, recommend items based on aggregated user preferences of those items, which does not depend on the availability of item descriptions. In CF, preference information from a set of users is utilised to make automatic predictions about the interests of the active user by assuming that user preferences hold over time. Importantly, predictions are made by models personalised to the taste of each active user based on information from many users, rather than from a global model making predictions for all users.

Collaborative filtering algorithms fall into two categories: memory-based and model-based approaches. Memory-based approaches [1,4,5,7] use heuristics to make rating predictions based on the entire collection of items previously rated by users. The unknown rating value $r_{c,s}$ of the active user c for an item s is typically computed as an aggregate of the ratings of users similar to c for the same item s. This aggregate can be an average or a weighted sum, where the weight is a distance that measures the similarity $sim(c_1, c_2)$ between users c_1 and c_2.

In contrast, model-based CF approaches [1,2,3,6,9] use the collection of ratings to learn a model, which is then used to make rating predictions. Although model-based methods have reported better accuracy of recommendation than memory-based approaches, there are also some limitations. Firstly, these methods are computationally expensive since they usually require all users and items involved to be used in creating models. Secondly, they attempt to predict the *rating* of a user rather than correctly *rank* the items.

1.2 People to People Recommendation

In this paper, we propose a recommendation method for people to people recommendation in social networks. In the traditional scenario where CF is applied, only the taste of users counts and items are passive in terms of the business transaction, i.e., once a user selects an item there is no response by that item. However in social networks, "items" are also users who actively participate in social interactions. In this sense, traditional CF is not applicable for people to people recommendation, since it only considers the taste of one side. We propose in this paper to extend traditional CF methods so the recommender system will handle the bilateral nature of such interactions.

We propose *SocialCollab*, a novel neighbour-based collaborative filtering algorithm to predict, for a given user, other users they may like to contact. This recommender system is based on the similarity of users in terms of the bilateral properties of attractiveness and taste. The main contribution of this paper is a novel approach for recommendation of potential friends or partners based on a new formalisation of the bilateral nature of interaction in social networks.

The paper is organised as follows. Section 2 presents a bilateral collaborative filtering framework for recommendation in social networks. Experimental evaluation is in Section 3 and conclusions are in Section 4.

2 Bilateral Collaborative Filtering

2.1 A Prototypical Collaborative Filtering Algorithm

Traditional collaborative filtering can operate in two directions: user-based or item-based. User-based approaches look for users who share the same rating patterns with the active user (the user whom the prediction is for) and then uses the ratings from like-minded users to calculate a prediction for the active user. On the other hand, item-based collaborative filtering such as that of Amazon.com [5] creates an item-item matrix determining relationships between pairs of items, which is then used to infer the taste of the active user.

The most important step in both approaches is determining similarity. Two items are *similar* if both are selected together by a set of users. Alternatively, two users are similar if they both select the same set of items (i.e., they have similar taste). The underlying assumption of CF approaches is that those who agreed in the past tend to agree again in the future. User-based approaches assume that two users will like the same items if they have similar taste. Therefore, an item is potentially recommended to the active user if it is selected by a similar user:

$$i \Rightarrow u : \exists s, (s \leftrightarrow u \wedge s \rightarrow i) \tag{1}$$

where $i \Rightarrow u$ denotes recommending i to u, $s \leftrightarrow u$ denotes that s is similar to u and $s \rightarrow i$ represents that s selected i.

Item-based approaches assume items can be related by the fact that they are frequently selected together by users, and will recommend an item which is similar to items that the active user selected:

$$i \Rightarrow u : \exists s, (s \leftrightarrow i \wedge u \rightarrow s) \tag{2}$$

These assumptions are only valid for recommending items to users where the selection is determined only by the user, not the item. In social networks, this is not the case – there is a two-way interaction. For user recommendation in social networks, collaborative filtering needs to be extended, as described in the next section.

2.2 Collaborative Filtering for Social Networks

In social networks, "items" as the recipients of actions are also *users* who are actively participating in social interactions. When they are contacted by other users, they can make different responses, either positive or negative. Therefore, traditional CF is not applicable to people recommendation since it only considers the taste of one side (users) and neglects the other (items). The recommender framework needs to be extended to handle the bilateral nature of such interactions in people recommendation.

Successful Interaction. We define an successful interaction as:

Definition 1. *An interaction between two users is a successful interaction when it has a positive response.*

Positive responses are usually defined in the application domain. For example, in an online dating site, a user Bob can send a message to another user Alice to express his interest in her. This message is a *contact.* This contact creates an *interaction* once it receives a corresponding *reply.* If the reply is positive, i.e. Alice also expresses her interest in Bob, this interaction becomes a successful interaction. Otherwise, it is an *unsuccessful interaction.*

User Attractiveness and Taste. In people recommendation, users have taste that determines their favourites when they actively make decisions selecting other users. At the same time, users are also passively involved in interactions by being selected by other users, which reflects, in some sense, their *attractiveness* within the social network. In this regard, both the aspects of users' taste and attractiveness need to be modelled. We define the similar attractiveness and similar taste of users as follows.

Definition 2. *Two users are similar in attractiveness ($u_i \overset{a}{\leftrightarrow} u_i$) if they are both selected by a nonempty set of users in common:*

$$u_i \overset{a}{\leftrightarrow} u_j : \exists U, (U \rightarrow u_i \wedge U \rightarrow u_j). \tag{3}$$

Definition 3. *Two users are similar in taste ($u_i \overset{t}{\leftrightarrow} u_j$) if they both select a nonempty set of users in common:*

$$u_i \overset{t}{\leftrightarrow} u_j : \exists U, (u_i \rightarrow U \wedge u_j \rightarrow U) \tag{4}$$

SocialCollab: Modelling Bilateral Decisions. Just because the active user likes a user does not mean a successful match since the liked user may not like the active user. This requires that the liked user also likes the active user. The point here is that only when u_a likes u_r and also u_r likes u_a can an interaction be a success. Only in this case, u_r should be recommended to u_a.

To model this behaviour, following traditional collaborative filtering assumptions, we define the following two assumptions based on user taste:

1. If people with similar taste to u_a like u_r, u_a will like u_r;
2. If people with similar taste to u_r like u_a, u_r will like u_a.

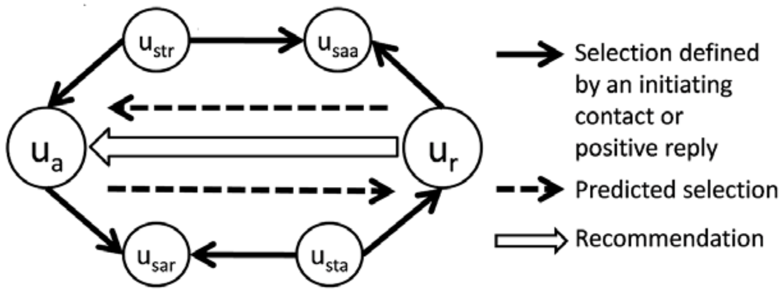

Fig. 1. SocialCollab recommender for bilateral collaborative filtering: u_a is the active user, u_r is the recommended user, u_{str} is a representative user with similar taste to the recommended user, u_{saa} is a representative user with similar attractiveness to the active user, u_{sar} is a representative user with similar attractiveness to the recommended user, and u_{sta} is a representative user with similar taste to the active user

This can be restated in terms of user attractiveness:

3. If u_a likes people with similar attractiveness to u_r, u_a will like u_r;
4. If u_r likes people with similar attractiveness to u_a, u_r will like u_a,

since both assumptions lead to the same predicted selections as illustrated in Figure 1.

Therefore, u_r should be recommended to u_a when u_r likes people with similar attractiveness to u_a and u_a likes people with similar attractiveness to u_r, or equivalently, when people with similar taste to u_r like u_a and people with similar taste to u_a like u_r.

More formally, for a predicted successful interaction between u_a and u_r: denoted $u_a \overset{*}{\rightarrow} u_r$, there are two conditions to be fulfilled:

5. The attractiveness of the recommended user should match the taste of the active user, which will facilitate initiation of the interaction from the active user to the recommended user. In a user-based approach, we define this as:

$$u_a \overset{*}{\rightarrow} u_r : \exists s, (s \overset{t}{\leftrightarrow} u_a \wedge s \rightarrow u_r) \tag{5}$$

   and its equivalence in terms of predicted selections in an item-based approach:

$$u_a \overset{*}{\rightarrow} u_r : \exists s, (s \overset{a}{\leftrightarrow} u_r \wedge u_a \rightarrow s) \tag{6}$$

6. The attractiveness of the active user should also match the taste of the recommended user, to ensure positive responses from the recommended user. In a user-based approach, this can be expressed as:

$$u_r \overset{*}{\rightarrow} u_a : \exists s, (s \overset{t}{\leftrightarrow} u_r \wedge s \rightarrow u_a) \tag{7}$$

   and its equivalence in an item-based approach:

$$u_r \overset{*}{\rightarrow} u_a : \exists s, (s \overset{a}{\leftrightarrow} u_a \wedge u_r \rightarrow s) \tag{8}$$

Algorithm 1. SocialCollab: Modelling Bilateral Decisions

```
Initialise C_{r,a} ← ∅; C_{a,r} ← ∅; R ← ∅
find users with similar taste (u_{str}, u_r)
find users with similar attractiveness (u_{sar}, u_r)
for all u_r do
  for all u_s : (u_s ∈ u_{str}) ∧ (u_s selects u_a) do
    C_{r,a} ← C_{r,a} ∪ {u_r}      // users with similar taste to u_r who selected u_a
  end for
end for
for all u_r do
  for all u_s : (u_s ∈ u_{sar}) ∧ (u_a selects u_s) do
    C_{a,r} ← C_{a,r} ∪ {u_r}      // users with similar attractiveness to u_r selected by u_a
  end for
end for
for all u_r : (u_r ∈ C_{r,a}) ∧ (u_r ∈ C_{a,r}) do
  R ← R ∪ {u_r}                    // recommendation set
end for
return R
```

Therefore, we have the following basis for people to people recommendation.

Definition 4. *A recommendation is a predicted successful interaction between two users:*

$$u_r \Rightarrow u_a : (u_r \xrightarrow{*} u_a \wedge u_r \xleftarrow{*} u_a) \tag{9}$$

Modelling User Selection We assume that if a user u_1 initiates an interaction by sending a contact to another user u_2, then u_1 likes u_2, which makes sense when considering people's interactions. However, initiating a contact is not the only way people can express their interest in others. If users receive contacts from others, they can also express their interest in the senders by sending positive responses back to the sender. Therefore we extend the model of user selection to include either initiating an interaction or giving a positive response to a contact initiated by another user.

Definition 5. *An extended selection between two users ($u_i \rightarrow u_j$) is a relationship:*

$$u_i \rightarrow u_j : (u_i \rightharpoonup u_j \vee u_i \rightharpoondown u_j) \tag{10}$$

where $u_i \rightharpoonup u_j$ means u_i initiates a contact to u_j and $u_i \rightharpoondown u_j$ indicates u_i responds positively to a contact from u_j.

The SocialCollab Algorithm. As depicted in Algorithm 1, the method works as follows. For each potential recommendation candidate u_r in the dataset, it first finds a set of users u_{str} having similar taste, and another set of users u_{sar} having similar attractiveness, to the candidate u_r. Then u_r is added to the recommendation set R for the active user u_a if at least one similar user in u_{str}

selects u_a and at least one similar user in u_{sar} is selected by u_a. The potential recommendations for user u_a are ranked according to the number of similar users in the set $C_{r,a} \cup C_{a,r}$.

3 Experimental Evaluation

In these experiments we aim to evaluate the proposed approach on people recommendation in a realistic setting. Therefore we applied our algorithm on a social network dataset from a commercial online dating site. We compare our learning algorithm SocialCollab to the standard CF algorithm. Data was pre-processed in Oracle 10 and algorithms were implemented in Matlab.

3.1 Experiment Setup

The datasets were collected from a commercial social network site containing interactions between users. Specifically, the data contains records each of which represents a contact as a tuple containing the identity of the contact's sender, the identity of the contact's receiver and an indicator showing whether the interaction was successful (with a positive response from the receiver to the sender) or unsuccessful (with a negative response).

The experiments were conducted on a training set covering a one week period and a test set on a subsequent week, both in March, 2009. Both training and test sets contain all users with at least one contact in the respective periods. The datasets used are summarised in Table 1.

Table 1. Dataset Description

	#Interactions	#Positive	#Negative	DSR	#U_a Involved
Training Set	188255	54754	133501	0.29	3746
Test Set	199083	56677	142406	0.28	2865

We compare SocialCollab to the standard CF algorithm using the evaluation metrics defined in the next section.

3.2 Evaluation Metrics

The evaluation metrics used in this research are defined as follows:

Definition 6. *Success Rate (SR) or Precision is the proportion of the true predicted successful interactions to all predicted successful interactions:*

$$SR = \frac{n_{tps}}{n_{ps}}, \tag{11}$$

where n_{tps} is the number of true predicted successful interactions and n_{ps} the number of predicted successful interactions.

Definition 7. *Default Success Rate (DSR) is the proportion of successful interactions to all interactions in the dataset:*

$$DSR = \frac{n_{ts}}{n_{all}}, \tag{12}$$

where n_{ts} is the number of true successful interactions and n_{all} the number of all interactions.

Definition 8. *Success Rate Improvement (SRI) is the ratio of success rate to the default success rate:*

$$SRI = \frac{SR}{DSR}. \tag{13}$$

Definition 9. *Recall is the proportion of the true predicted successful interactions to all true successful interactions:*

$$Recall = \frac{n_{tps}}{n_{ts}}, \tag{14}$$

where n_{tps} is the number of true predicted successful interactions and n_{ts} the number of successful interactions in the dataset.

3.3 Results of Recommendation

We compare SocialCollab to the standard collaborative filtering CF and its extended version CF+ using the proposed selection method defined in Definition 5. More specifically, CF+ uses the extended selection of Definition 5 rather than ordinary selection as used in standard CF. The details of the comparison results of those algorithms on the Top 100 and Top 1000 are shown in Tables 2 and 3, which shows that the proposed algorithms SocialCollab and CF+ both outperform the standard CF for recommendation, with the SocialCollab the best performer. As shown in Table 4, SocialCollab achieves approximately 0.35 SR on average for the Top 100 recommendations for each active user. This gives an SRI of about 1.25. The SRI for CF+ and CF on the Top 100 are less than 1 because the majority of interactions in the dataset are negative, leading to many predicted interactions that are unsuccessful. Figure 2 shows that CF performs at around the default, and CF+ performs better than CF. SocialCollab outperforms CF+.

Table 2. Comparison on SR for Top 100 Recommendations

Top	10	20	30	40	50	60	70	80	90	100
SocialCollab	0.35	0.34	0.34	0.35	0.34	0.35	0.35	0.35	0.35	0.35
CF+	0.30	0.28	0.27	0.27	0.28	0.28	0.27	0.27	0.26	0.26
CF	0.28	0.27	0.26	0.26	0.26	0.26	0.26	0.26	0.25	0.25

Table 3. Comparison on SR for Top 1000 Recommendations

Top	100	200	300	400	500	600	700	800	900	1000
SocialCollab	0.35	0.36	0.36	0.37	0.37	0.37	0.37	0.37	0.37	0.37
CF+	0.26	0.27	0.27	0.28	0.28	0.28	0.28	0.28	0.28	0.28
CF	0.25	0.26	0.26	0.27	0.27	0.27	0.27	0.27	0.27	0.27

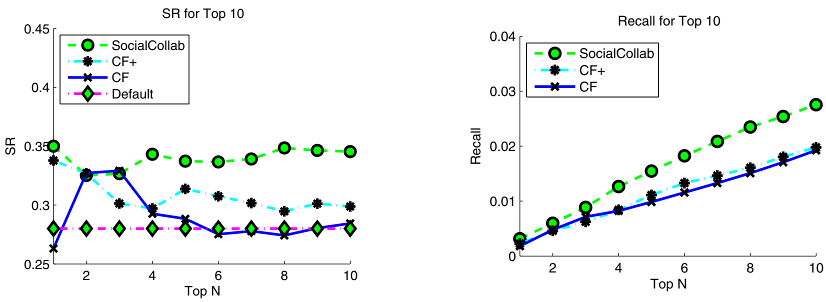

Fig. 2. Comparisons of SR (left) and Recall (right) for Top 10

Table 4. Comparison of Ranked Recommendation Results

		SR		SRI	
		Top 100	Top 10	Top 100	Top 10
A	SocialCollab	0.35	0.35	1.25	1.25
B	CF+	0.26	0.30	0.93	1.07
C	CF	0.25	0.28	0.89	1
D	Default	0.28	0.28	1	1
	Impvt. of A over B	0.09	0.05	0.57	0.18
	Impvt. of A over C	0.10	0.07	0.36	0.25
	Impvt. of A over D	0.07	0.07	0.25	0.25

4 Concluding Remarks

We have proposed an approach for people recommendation by collaborative filtering. Our experimental results show that the novel SocialCollab recommender performs well in people to people recommendation on social network data from a commercial online dating site. The proposed algorithms SocialCollab and CF+ both outperform standard CF as measured on both Precision (SR) and Recall, with SocialCollab being the best. A general framework for ranking in the context of the SocialCollab algorithm is the subject of further work.

Acknowledgement

This project is funded by the Smart Services CRC under the Australian Government's Cooperative Research Centre program.

References

1. Breese, J.S., Heckerman, D., Kadie, C.: Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In: Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, pp. 43–52 (1998)
2. Getoor, L., Sahami, M.: Using Probabilistic Relational Models for Collaborative Filtering. In: Working Notes of the KDD 1999 Workshop on Web Usage Analysis and User Profiling (1999)
3. Hofmann, T.: Collaborative Filtering via Gaussian Probabilistic Latent Semantic Analysis. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 259–266 (2003)
4. Konstan, J., Miller, B., Maltz, D., Herlocker, J., Gordon, L., Riedl, J.: GroupLens: Applying Collaborative Filtering to Usenet News. Communications of the ACM 40(3), 77–87 (1997)
5. Linden, G., Smith, B., York, J.: Amazon. com Recommendations: Item-to-Item Collaborative Filtering. IEEE Transactions on Internet Computing 7(1), 76–80 (2003)
6. Pavlov, D., Pennock, D.M.: A Maximum Entropy Approach to Collaborative Filtering in Dynamic, Sparse, High-Dimensional Domains. In: Neural Information Processing Systems, pp. 1441–1448 (2002)
7. 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 1995), pp. 210–217 (1995)
8. Su, X., Khoshgoftaar, T.: A Survey of Collaborative Filtering Techniques. In: Advances in Artificial Intelligence 2009 (2009)
9. Ungar, L., Foster, D.: Clustering Methods for Collaborative Filtering. In: Proceedings of the AAAI 1998 Workshop on Recommender Systems, pp. 112–125 (1998)