

# Incorporating Multiple Attributes in Social Networks to Enhance the Collaborative Filtering Recommendation Algorithm

Jian Yi

Chongqing University of Posts and  
Telecommunications  
Chongqing, China  
Chongqing Engineering Laboratory  
of Network and Information Security

Xiao Yunpeng

Chongqing University of Posts and  
Telecommunications  
Chongqing, China  
Chongqing Engineering Laboratory  
of Network and Information Security

Liu Yanbing

Chongqing Engineering Laboratory  
of Network and Information Security  
Chongqing University of Posts and  
Telecommunications  
Chongqing, China

**Abstract**—In view of the existing user similarity calculation principle of recommendation algorithm is single, and recommender system accuracy is not well, we propose a novel social multi-attribute collaborative filtering algorithm (SoMu). We first define the user attraction similarity by users' historical rated behaviors using graph theory, and secondly, define the user interaction similarity by users' social friendship which is based on the social relationship of being followed and following. Then, we combine the user attraction similarity and the user interaction similarity to obtain a multi-attribute comprehensive user similarity model. Finally, realize personalized recommendation according to the comprehensive similarity model. Experimental results on Douban and MovieLens show that the proposed algorithm successfully incorporates multiple attributes in social networks to recommendation algorithm, and improves the accuracy of recommender system with the improved comprehensive similarity computing model.

**Keywords**—Recommender System; Social Networks; Collaborative Filtering; Comprehensive Similarity

## I. INTRODUCTION

Social networks and recommender system are quickly becoming popular. Collaborative filtering is treated as a technique in assisting users to locate what they are interested in a timely manner [1]. Collaborative relationships in recommender systems can be represented as a social network [2], the growth of social networks and the development of personalized recommendation techniques have evidently improved users' experiences and delivered higher quality of services [3]. However, social recommender systems are significantly challenged by the data sparsity issue -- the social network topology structure shows that only a small number of users have relatively many connections with other users, and most of the users have very few or no connections. In other words, the number of users and the fan relationship follow the long tail distribution [4, 5, 6]. Also, users sharing similar interests in social networks generally have a tendency to contact with each other [7]. Traditional personalized recommendation methods fail to take into account users' social relationships and the fact that a user's interests may be affected by another user's interests through the social relationship, resulting in inferior recommendation quality.

The objective of this paper is to propose a new comprehensive similarity model to determine neighbors set and top-N items list recommended to the target user, thereby making a new contribution towards the solution of the data sparsity problem. Experiments of the approach were expounded and proved on both MovieLens and Douban dataset.

In short, the main contributions of the paper can be summarized as follows: A new comprehensive similarity measurement is proposed by devising and integrating user attraction similarity and user interaction similarity within a friend-user-item framework.

The newly proposed method outperforms some of the peer collaborative filtering algorithms.

## II. RELATED WORK

Extensive researches have been done in terms of dealing with both the data sparsity issue in collaborative filtering and the fact that traditional recommendation methods are not directly suitable for social networks.

The most frequently-used recommender systems are collaborative filtering [8, 9], they analyze users behavior in the past and mine correlations between users and items. The similarity calculation of items or users is a core problem of collaborative filtering. In [10] and [11], both users and items are considered in the determination of user similarity to improve the prediction accuracy. In [12], researchers incorporate user-based and item-based methods to reduce the computational costs. Huang presented a graph-based approach in which users' tastes are assumed to be "transitive", this approach enhances the information matrices and thereby contributes to the resolution of the data sparsity problem [13].

Social recommendation algorithms are also very popular in order to address limitation of the collaborative algorithms caused by the data sparsity. The research of [14] fuses the collaborative filtering and social network information into one model and the ensuing strategy is able to dynamically adjust the weight of each attribute resulting in a notable balance in terms of accuracy and coverage. Liu proposed a user similarity model which effectively improves the quality of recommender

systems by combing users' Weibo contents, social networks, and users' activities on Weibo [15]. Also, Roth suggested and verified an improved friend recommender system which generates groups of friends by mining implicit graphs in social networks and reportedly leads to an increase in user satisfactions for recommender systems [16]. Konstas effectively integrated social networks and the recommender system by studying additional relationships [17], and as social community discovery algorithm [18, 19, 20], Bayesian personalized ranking model [21] are proposed with social information aiming to enhance the accuracy of recommendations. There are also studies [22, 23] in the literature that investigate the roles played by social elements such as friendship and trust in the context of collaborative filtering. Surprisingly, it is interesting to note that social network information based recommendation methods may excel mathematical algorithms [24]. Rong studied how to predict a user's social connections by means of some public data involved in e-mails for purpose of recommending friends for users [25]. No matter what needs to be recommended, friends or items, it seems certain that the social network information is finding wider and deeper applications in the construction of recommender systems.

### III. INCORPORATING SOCIAL MULTIATTRIBUTE FOR RECOMMENDERS

#### A. Problem Description

The selection of the nearest neighbors set for the target user is the critical task of the collaborative filtering method, and is always determined by the similarity among users. We assume that users' behavior dataset and users' social network information dataset are both available, where the former describes users' various behaviors and interests in the past, and the latter shows the following and/or being followed relationships among users.

We consider both interest similarity and social connections of users, and propose the notion of comprehensive similarity, which is defined by combining users' attraction similarity with users' interaction similarity.

The attraction similarity is a measurement of users' likeness in terms of their interests. Two users sharing the same kind of taste on most items with a small interest-gap (defined formally in Section 3.3.1) will have a high attraction similarity. In addition to the attraction similarity, we also define the interaction similarity among users to measure their interactive similarity degree in social networks.

Since users in general tend to trust more on items recommended by friends with high interaction similarities, this mechanism may provide new users who do not have any user behaviors with some high quality recommendations thereby alleviating the data sparsity problems. As such, the comprehensive similarity delivers a more accurate measurement of the analogy between users and indicates users' interests more precisely.

#### B. Algorithmic Framework

The proposed algorithm SoMu is based on the memory-based collaborative filtering algorithm. According to regular

procedure of collaborative filtering, note that various types of information can be used to calculate the user similarity in social networks. We choose to use the information of users ratings on items to devise the attraction similarity, and the information about users interdependency in social networks to devise the interaction similarity. SoMu completes its task by the following steps.

Step No.1 is to collect the available data including social relationship, user profiles, and item profiles. And social relationship contains all users following and followed relationship. User profiles contain user id, user preference and so on, and item profiles contain item id, item association attributes and so on.

Step No.2 is to clean the above data and develop the user node adjacency matrix and item rating matrix. The two cleaned matrix respectively represent users all neighbors and all the cleaned ratings. These basic data are sources of the following similarities.

Step No.3 is to calculate attraction similarity for all users, complying with the formula defined in section 3.3.1.

Step no.4 is to calculate interaction similarity for all users, complying with the formula defined in section 3.3.2.

Step No.5 is to combine attraction similarity and interaction similarity to get the comprehensive similarity, which is the measure of dividing neighbors set.

Step No.6 is to output the personal top n recommendations for all target users. As shown in the dotted line, all the final results would react to user profiles and item profiles for further research. Figure 1 depicts the algorithmic framework of this approach.

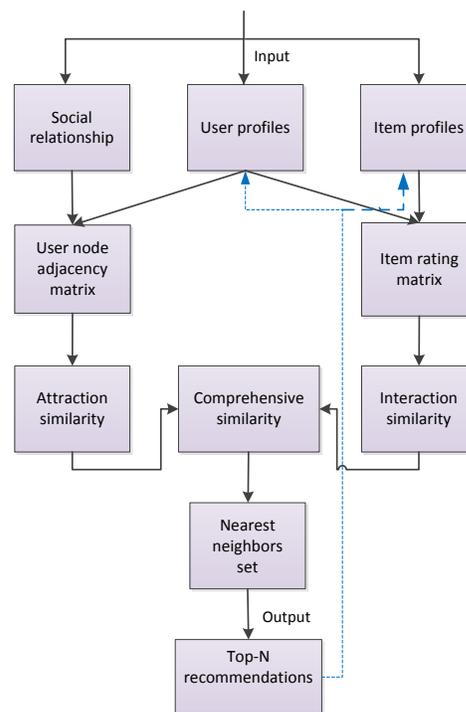


Fig. 1. Framework of the proposed algorithm SoMu

### C. Comprehensive User Similarity Model

As indicated in Figure 2 (where capital letters in circles represent users and lower case letters in squares represent items), the proposed comprehensive similarity model is obtained by considering and integrating the followed-following relationship among users and the rating relationship among users and items. Specifically, the comprehensive similarity  $W(u, v)$  between two users  $u$  and  $v$  is calculated as follows:

$$W(u, v) = \alpha * W(u, v)_{att} + \beta * W(u, v)_{int} \quad (1)$$

Where  $W(u, v)_{att}$  and  $W(u, v)_{int}$  denote the attraction similarity and the interaction similarity between and the users  $u$  and  $v$ , respectively. Also,  $\alpha$  and  $\beta$  are weights satisfying  $\alpha + \beta = 1$ .

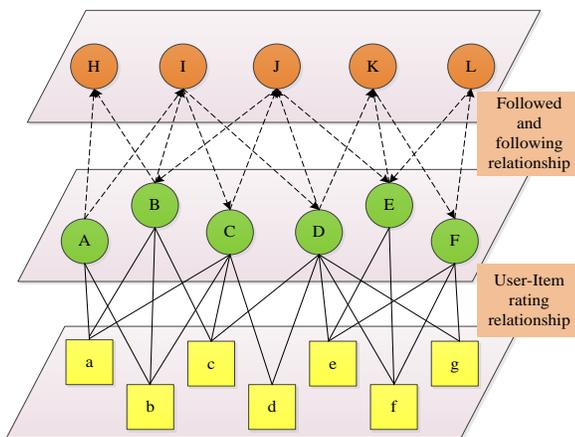


Fig. 2. The model of the comprehensive user similarity

#### Attraction Similarity

In user attraction similarity, the distance called interest-gap between two users is measured by their ratings on common items. A smaller interest-gap indicates a greater attraction that common items have on these two users. We describe the construction of the user attraction similarity. Considering that users and items in social networks can be regarded as nodes in graphs, weighted bipartite graphs are a natural choice for modeling the behaviors of users with respect to items.

A weighted bipartite graph is a 4-tuple  $(U, I, E, w)$ , where  $U$  is the set of user nodes,  $I$  is the set of item nodes,  $E$  is the set of edges connecting user nodes and item nodes, and  $w: E \rightarrow Z^+$  is a function from  $E$  to the set  $Z^+$  of positive integers. If user (node)  $u$  has a rating for item (node)  $i$ , then there would be an edge  $e \in E$  connecting node  $u$  and node  $i$ , and  $w(e)$  would be the value of the rating. For example, Figure 3 shows the situation where the user set is  $\{A, B, C\}$  and the item set is  $\{a, b, c, d\}$  with A having ratings for  $\{a, b, c\}$ , B for  $\{b, d\}$ , and C for  $\{b, c, d\}$ .

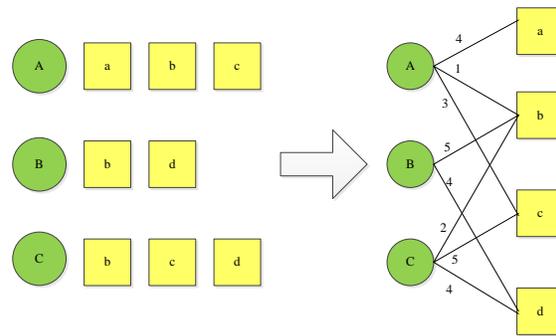


Fig. 3. User-Item rating relationship represented by a weighted bipartite graph

The attraction similarity  $W(u, v)_{att}$  between users  $u$  and  $v$  is computed as follows. Let  $N(u)$  be the set of items which user  $u$  has rated,  $M(i)$  be the set of users who has rated the item  $i$ ,  $r_{max}$  be the maximum possible rating on items,  $r_{min}$  be minimum possible rating on items, and  $\gamma$  be the normalized factor. Then, for any pair of users  $u$  and  $v$ , when  $N(u) \cap N(v) \neq \emptyset$ ,

$$W(u, v)_{att} = \gamma \cdot \frac{\sum_{i \in N(u) \cap N(v)} \frac{1}{\log(1 + |M(i)|)}}{R(u, v)^2}$$

$$R(u, v) = \frac{A + B}{|N(u) \cap N(v)|}$$

$$A = \sum_{\substack{i \in N(u) \cap N(v) \\ r_{ui} \neq r_{vi}}} (r_{max} - \min(r_{ui}, r_{vi}) + r_{min})$$

$$B = \sum_{\substack{i \in N(u) \cap N(v) \\ r_{ui} = r_{vi}}} r_{min}$$

(2)

otherwise,

$$W(u, v)_{att} = 0.$$

(3)

#### Interaction Similarity

For popular commodities, we assign a penalty parameter to them to adjust the calculation since users will naturally have a high rating on popular commodities. The interaction similarity is computed by considering a target user's follower set and the set of users that this target user follows in a cosine-like setting. Note that the former set is the set of users who actively make friends with this target user, and the latter set is the set of users with whom this target user makes friends. Similar to that the user-item rating relationship can be modeled by weighted bipartite graphs, users following-followed relationship can be modeled by directed graphs. If user  $u$  follows user  $v$ , there would be an arrow from user node  $u$  to user node  $v$  in the graph. For any user  $u$ , we use  $out(u)$  to denote the set of users whom user  $u$  follows, and  $in(u)$  to denote the set of

users who follow user  $u$ . In other words,  $out(u)$  represents the connections that user  $u$  has, and  $in(u)$  represents the influences that user  $u$  has exerted. Figure 4 shows the following-followed relationship among users  $A, B, \dots, K$ , where user  $B$  follows users  $D, E$  and  $G$ , and user  $J$  is followed by users  $D, F$  and  $G$ . That is,  $out(B) = \{D, E, G\}$  and  $in(J) = \{D, F, G\}$ .

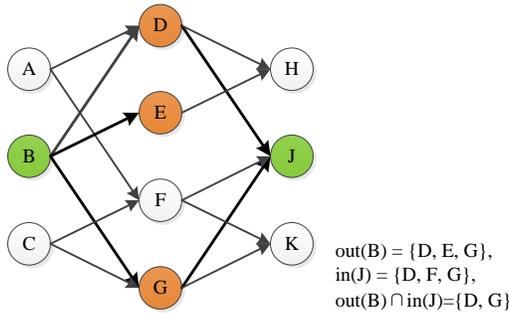


Fig. 4. Users' following-followed relationship model

Based on the observation that users tend to trust more on items recommended by friend that they follow,

$$W(u, v)_{int} = \frac{|out(u) \cap in(v)|}{\sqrt{|out(u)| |in(v)|}} \quad (4)$$

we define the interaction similarity  $W(u, v)_{int}$  between two users  $u$  and  $v$  to be formula 4.

#### D. Top-N Recommendations

In recent years, researches show that forecasting the items that will attract the user is more meaningful than predicting about what scores the user will rate on items. That is, the top-N prediction may be considered more valuable than the score prediction. Many existing recommendation algorithms are based on top-N prediction, and have great performance [26, 27]. We take the top-N approach in this paper. To generate a top-N item list for a user  $u$ , we calculate a predicting score  $p_{ui}$  for each candidate item  $i$  as follows:

$$p_{ui} = \sum_{v \in N(i) \cap S(u, k)} W(u, v) * r_{vi} \quad (5)$$

where  $S(u, k)$  is the set of  $k$  nearest neighbors of user  $u$  which is decided by the integrated similarity of  $u$  to other users, and then rank items according to the scores. The algorithm for the top-N recommendation, named SoMu, is shown below. (see Algorithm 1).

This algorithm computes the attraction similarity by setting up a user-item reversal list first and then constructing a matrix  $W$  of size  $|U| \times |U|$  which will be used as the numerator in the computation of the attraction similarity.

#### ALGORITHM 1. SoMu collaborative filtering algorithm

##### Algorithm SoMu Collaborative Filtering Algorithm

**Input:**  $M_{rating}, R_{follow}, K, N$

**Output:**  $L_{top-N}$

**Procedure begin**

- 1: Define bipartite graph mode  $G(U, I, E, w)$  by the user-item rating matrix  $M_{rating}$ ;
- 2:     **for**  $u, v \in G$  and  $u \neq v$  **do**
- 3:         Compute  $W(u, v)_{att}$ ;
- 4:     **end for**
- 5: Define the out-degree matrix and in-degreed matrix  $O$  and  $I$  by the user following and followed-matrix  $R_{follow}$ ;
- 6:     **for**  $u \in O, v \in I$  and  $u \neq v$  **do**
- 7:         Compute  $W(u, v)_{int}$ ;
- 8:     **end for**
- 9: Generate  $W(u, v)$  as the liner combination of  $W(u, v)_{att}$  and  $W(u, v)_{int}$ ;
- 10: Compute the  $K$  nearest neighbors by  $W(u, v)$ , then compute recommendation list  $L_{top-N}$  of length  $N$ ;
- 11:     **return**  $L_{top-N}$ ;

**Procedure end**

$W[u][v]$  and  $W[v][u]$  will be incremented by 1 if users  $u, v$  both rate an item  $a$ . Iterating through the list of all items will give rise to the matrix  $W$ . For the denominator of the attraction similarity computation, an interest-difference gap matrix  $R$  is constructed by using a hash table which is of  $O(1)$  time.

The entire computation of the attraction similarity costs time  $O(n^2)$  where  $n$  is the number of nodes in  $U$ . Although the time complexity of SoMu is general, it plays well on evaluation metrics such as precision, recall, coverage, and popularity, which would be mentioned farther below.

#### E. Identify the Headings

We use the following metrics to evaluate the quality of the proposed top-N recommendation algorithm described in the previous section: *Precision*, *Recall*,  $F_{PR}$ , *Coverage* and *Popularity*.

In all formulas below,  $R(u)$  represents the set of the top-N items recommended for user  $u$ ;  $T(u)$  represents the set of items that user  $u$  actually rates in a testing bed;  $U$  is the set of users; and  $I$  is the set of items. *Precision* and *Recall* can be seen as a pair of quality assessment for information retrieval [30], and the same for recommendations. Figure 5 shows the original definition of precision and recall.

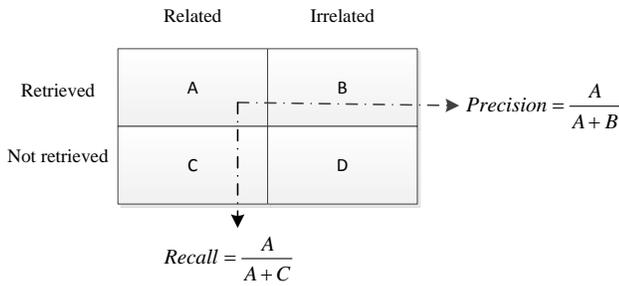


Fig. 5. Precision and Recall description

Considering that *Precision* and *Recall* are individual measurements and are related to each other, we devise another measurement  $F_{PR}$  as follows to indicate the effects of both *Precision* and *Recall*.

$$Precision = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |R(u)|}$$

$$Recall = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |T(u)|}$$

(6)

$$F_{PR} = \frac{(\lambda^2 + 1) Precision * Recall}{\lambda^2 (Precision + Recall)}$$

(7)

In (7),  $\lambda$  is weight, usually  $\lambda = 1$  is frequently-used. In addition to *Precision*, *Recall* and  $F_{PR}$  which evaluate the accuracy of the recommendation algorithm, the notion of *Coverage* is used to indicate the long-tail exploration capability of the algorithm. In [33], data from social network was analyzed in order to find what information could improve the diversity and coverage of recommendations. the notion of *Coverage* is used to indicate the long-tail exploration capability of the algorithm.

$$Coverage = \frac{|\bigcup_{u \in U} R(u)|}{|I|}$$

(8)

Finally, *Popularity* is used to indicate whether the recommendation results are new. A smaller *Popularity* value means that most of the recommended items are not very popular and suggests that the algorithm works better. The definition of *Popularity* is given as follows [29].

$$Popularity = \frac{\sum_{u \in U} \sum_{i \in T(u)} \log(1 + |M(i)|)}{\sum_{u \in U} |T(u)|}$$

(9)

#### IV. EXPERIMENTS

We in this section present the experimental testing result for the algorithm SoMu discussed in the previous section.

#### A. Datasets

We tested the SoMu algorithm on two publicly available datasets: MovieLens (<http://www.grouplens.org>) and Douban (<http://datatang.com>). The practical scale of experimental datasets are shown below TABLE.1.

TABLE I. EXPERIMENTAL DATA SCALE

dataset	user	item	rating	relationship
Douban	38303 3	80008	3648104	100000
MovieLens	934	1682	100000	\

Note that the following-followed social links in Douban dataset is unidirectional and thus can be understood as the in-degree and out-degree of the user nodes in directed graphs (as we discussed in the previous section). While a user's in-degree is the indicator of his/her social status and influences, a user's out-degree is the indicator of the number of other users that he/she cares and follows. It can be seen clearly that both users' in-degrees and out-degrees are in line with the long-tail distribution.

#### B. Design of the Experiment

We randomly divide the set of the user behavioral data into two parts as follows: 80% of the data is used as the training set and 20% of the data is used as the testing set. The algorithm is applied to the training set to obtain the top-N recommendation list for the user and is used on the testing set for the purpose of performance evaluation. Specifically, we set up the following three experiments:

- Using the MovieLens 100k dataset, compare SoMu algorithm with a peer collaborative filtering algorithm to see which one has a higher comprehensive measurement  $F_{PR}$ .
- Using the dataset from Douban, observe the performance of the SoMu algorithm to determine if the addition of social information into a collaborative filtering algorithm can improve the quality of the algorithm. If so, go ahead and pursue the values of the parameters  $K$ ,  $N$ ,  $\alpha$ ,  $\beta$  which may enable the best performance of the algorithm.
- Based on the result of experiment 2, compare the performance of SoMu with that of another peer social recommendation algorithm in terms of the recommendation quality.

#### C. Experimental Results

**Experiment 1.** In this experiment, compare the algorithm SoMu with one of the traditional collaborative filtering algorithms UserCF [11] in terms of the comprehensive evaluation metric  $F_{PR}$ . The comparison result is shown in Figure 6 with  $\lambda = 1$  and number of neighbors ranging from 5 to 100. Given the fact that a larger  $F_{PR}$  indicates a superior algorithm, it can be seen clearly that the proposed algorithm outperforms UserCF, and would be a preferred choice when a special requirement such as finding an equilibrium between *Precision* and *Recall* is needed.

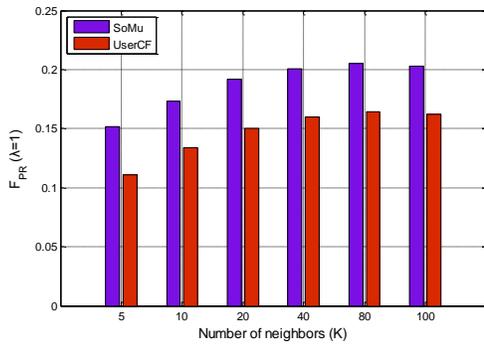


Fig. 6. Comparison between SoMu and UserCF with respect to  $F_{PR}$

**Experiment 2.** In order to see the effectiveness (or non-effectiveness) of different similarities on the evaluation metrics *Precision*, *Recall*,  $F_{PR}$ , *Coverage* and *Popularity*, we in this experiment conduct three sub-experiments: (1) SoMu@1: implement the recommendation solely by the attraction similarity, (2) SoMu@2: implement the recommendation solely by the interaction similarity, and (3) SoMu@3: implement the recommendation by the combination of the attraction similarity and the interaction similarity.

Since the data sparsity of the user rating matrix for items may affect the result of recommendations, we conduct the experiments on the basis of various instances of data sparsity. In order to give a quantitative measurement for data sparsity, define the notion of user sparsity as follows:

$$D_{u\_spa} = S(u) \tag{10}$$

Where  $u$  denotes a user and  $S(u)$  denotes the number of rated items by user  $u$ . In the experiments,  $D_{u\_spa}$  is set to be many different values with any two consecutive values differentiated by 30.

Also, note that experimental results can be affected by the following parameters:  $K$ ,  $N$ ,  $\alpha$ ,  $\beta$ . (Recall that  $K$  denotes the number of the nearest neighbors to the target user in the process of recommendation,  $N$  is the length of the recommendation list, and  $\alpha$  and  $\beta$  are weight factors in the computation of the comprehensive similarity.) As such, different experiments are devised to examine these possible and potential impacts. Specifically, algorithm SoMu@1 corresponds to the computation of formula (1) with  $\alpha \neq 0$  and  $\beta = 0$ , and thus is completely determined by the attraction similarity; algorithm SoMu@2 corresponds to the computation of formula (1) with  $\alpha = 0$  and  $\beta \neq 0$ , and thus is completely determined by the interaction similarity; algorithm SoMu@3 corresponds to the computation of formula (1) with  $\alpha \neq 0$  and  $\beta \neq 0$ , and thus is completely determined by the integrated similarity proposed in this paper. Figures 7 – 10 show the comparisons of SoMu@1, SoMu@2, and SoMu@3 in terms of *Precision*, *Recall*, *Coverage* and *Popularity*.

All experiments shown in Figures 7-10 are conducted for some given and fixed user sparsity. Figure 7 illustrates the correlation between  $K$  and *Precision*; see that SoMu@3 outperforms SoMu@1 slightly, but beats SoMu@2 to a large extent. Figure 8 shows the correlation between  $K$  and *Recall*. Again, we are able to observe that SoMu@3 is superior to both SoMu@1 and SoMu@2. Figure 9 exhibits the correlation between  $K$  and *Coverage*, and clearly indicates that SoMu@3 has a stronger long-tail item mining capability than both SoMu@1 and SoMu@2. Figure 10 depicts the correlation between  $K$  and *Popularity*. A low *Popularity* of a recommender system means that the items recommended by this system are not those hot, popular commodities on the market, which indicates that this recommender system has a certain degree of novelty. A high *Popularity* of a recommender system would mean the opposite. In Figure 10, that the *Popularity* of SoMu@3 is lower than that of SoMu@1 but higher than that of SoMu@2, resulting in a balanced state in terms of recommendation novelty and item recognitions.

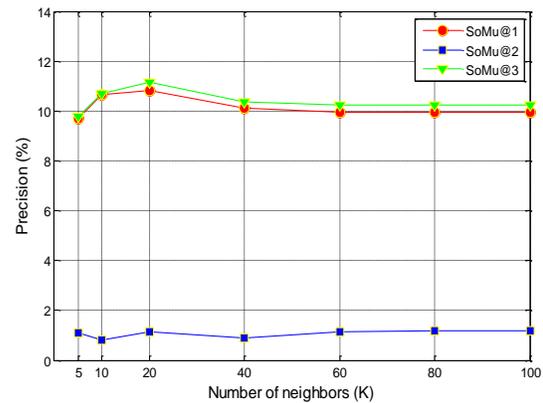


Fig. 7. Correlation between K and Precision

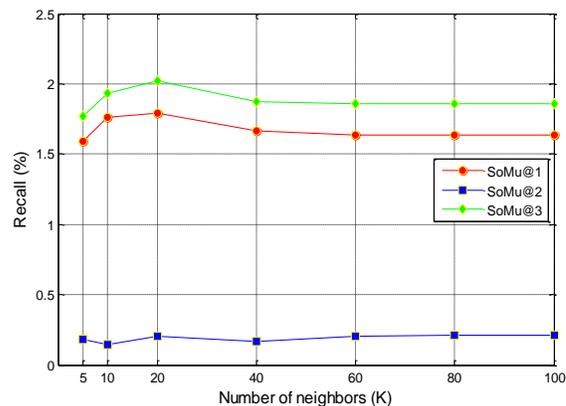


Fig. 8. Correlation between K and Recall

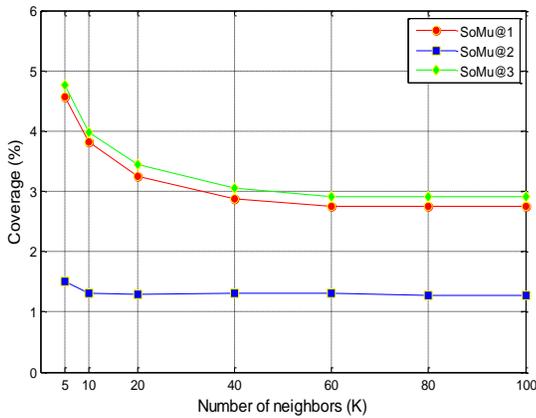


Fig. 9. Correlation between K and Coverage

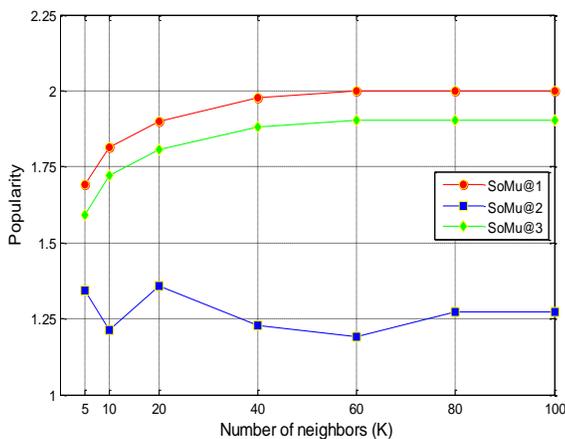


Fig. 10. Correlation between K and Popularity

Figures 7-10 clearly and heuristically indicate that the algorithm tends to be stable for all aspects when  $K$  is sufficiently large, although the rigorous such argument needs to be proved mathematically. Also, we can determine by these figures that the optimal values for  $K, N, \alpha, \beta$  are  $K=20, N=24, \alpha=0.988, \beta=0.012$ .

**Experiment 3.** In this experiment, we compare the SuMo algorithm with one of the typical social recommendation algorithms Neighbor [29]. The typical algorithm used Pearson correlation to calculate user similarity. Pearson correlation is defined as below, and the meaning of the symbols and letters is the same as above formulas, and need not be repeated here.

$$w_{ij} = \frac{\sum_{u \in U} (r_{ui} - \bar{r}_i)(r_{uj} - \bar{r}_j)}{\sqrt{\sum_{u \in U} (r_{ui} - \bar{r}_i)^2 \sum_{u \in U} (r_{uj} - \bar{r}_j)^2}} \quad (11)$$

Based on Experiment 2, set the values of parameters in SuMo to the optimal ones, i.e.,  $K=20, N=24, \alpha=0.988, \beta=0.012$ . The comparison results are depicted in Figures 11 and 12 and TABLE 2.

TABLE II. COMPARISON BETWEEN SUMO AND NEIGHBOR ALGORITHMS WRT PRECISION AND RECALL

K	Neighbor		SoMu	
	Precision	Recall	Precision	Recall
5	9.378%	1.759%	9.776%	1.769%
10	10.652%	1.927%	10.684%	1.934%
20	<b>11.111%</b>	<b>2.011%</b>	<b>11.165%</b>	<b>2.021%</b>
40	10.256%	1.856%	10.363%	1.876%
60	10.150%	1.837%	10.256%	1.856%
80	10.150%	1.837%	10.256%	1.856%
100	10.150%	1.837%	10.256%	1.856%

The data in Table 2 clearly show that SoMu exceeds Neighbor in terms of performance metrics *Precision* and *Recall*, and also show that both SoMu and Neighbor reach their own best performance at the optimal parameter setting ( $K = 20$ ). Figure 11 demonstrates the comparison between SoMu and Neighbor with respect to evaluation metric *Coverage*. Evidently, SoMu outperforms Neighbor in *Coverage* although the two methods' performances trend in the same manner. The comparison between SoMu and Neighbor in regards to *Popularity* is given in Figure 12. We are able to note that SoMu has a higher *Popularity* than Neighbor prior to the stabilization of these two algorithms, indicating that SoMu, during this period of time, primarily recommends recognized and fashionable items to the users. However, as the algorithm tends to stabilize with the increase of  $K$ , SoMu exhibits a lower *Popularity* than Neighbor, indicating that SoMu starts to recommend non-fashionable items to the user with a sense of novelty.

In summary, Table 2, Figure 11, and Figure 12 suggest that the proposed algorithm SoMu outperforms the algorithm Neighbor in terms of all evaluation metrics. Also, we can see that all evaluation metrics tend to become a constant as the number of neighbors  $K$  is sufficient large.

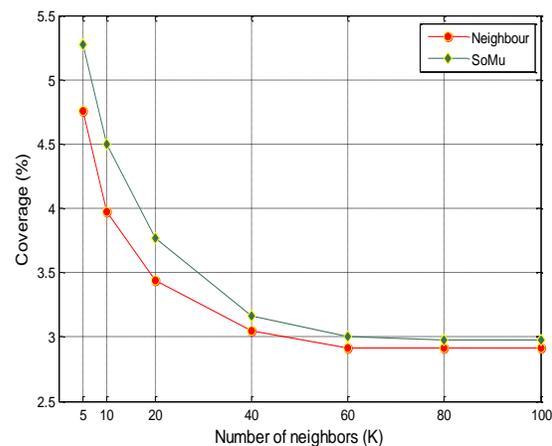


Fig. 11. Comparison between SuMo and Neighbor Algorithms wrt Coverage

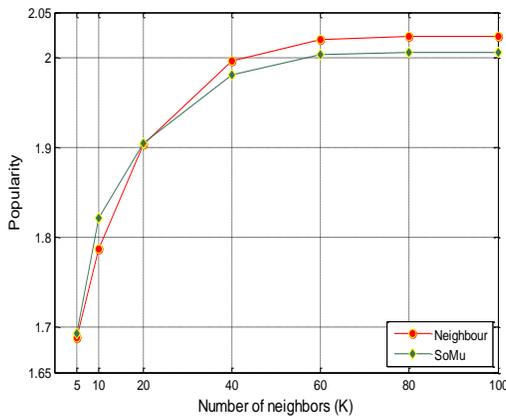


Fig. 12. Comparison between SuMo and Neighbor Algorithms wrt Popularity

## V. CONCLUSIONS

We have in this paper proposed a new collaborative filtering recommendation algorithm SoMu which leverages multiple attributes in social networks to improve the recommendation result. By applying proposed to the popular datasets obtained on MovieLens and Douban and comparing the outcomes with that obtained from other peer recommendation algorithms, we have found that SoMu excels other peer algorithms in terms of recommendation evaluation metrics. As the further work, we plan to deepen the study on the correlations between recommender systems and the social networks by further investigating the relations formed among various groups on the social networks and by associating items recommendations with friend recommendations. We plan to parallelize the algorithm, and increase the amount of experimental data.

## REFERENCES

- [1] G Adomavicius, A Tuzhilin. "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions". Knowledge and Data Engineering, IEEE Transactions on, 2005, 17(6): 734-749.
- [2] R Burke, M P O'Mahony, N J.Hurley "Robust collaborative recommendation". Recommender Systems Handbook. Springer US, 2011: 805-835.
- [3] P Kazienko, K Musial, T Kajdanowicz. "Multidimensional social network in the social recommender system". Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, 2011, 41(4): 746-759.
- [4] H J Ahn. "A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem". Information Sciences, 2008, 178(1): 37-51.
- [5] Y J Park, A Tuzhilin. "The long tail of recommender systems and how to leverage it". Proceedings of the 2008 ACM conference on Recommender systems. ACM, 2008: 11-18.
- [6] Y Moshfeghi, B Piwowarski, J M Jose. "Handling data sparsity in collaborative filtering using emotion and semantic based features". Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. ACM, 2011: 625-634.
- [7] J Odonovan, B Smyth. "Trust in recommender systems". Proceedings of the 10th international conference on Intelligent user interfaces. ACM, 2005: 167-174.
- [8] C Desrosiers, G Karypis. "A comprehensive survey of neighborhood-based recommendation methods". Recommender systems handbook. Springer US, 2011: 107-144.

- [9] Y Koren, R Bell. "Advances in collaborative filtering". Recommender systems handbook. Springer US, 2011: 145-186.
- [10] W Jun, A P De Vries, M J T Reinders. "Unifying user-based and item-based collaborative filtering approaches by similarity fusion". Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2006: 501-508.
- [11] K Zhao, L Pengyu. "Improved collaborative filtering approach based on user similarity combination". Management Science & Engineering (ICMSE), 2014 International Conference on. IEEE, 2014: 238-243.
- [12] L Qingwen, X Yan, H Wencho. "Combining User-Based and Item-Based Models for Collaborative Filtering Using Stacked Regression". Chinese Journal of Electronics, 2014, 23(4).
- [13] H Zan, H Chen, D Zeng. "Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering". ACM Transactions on Information Systems (TOIS), 2004, 22(1): 116-142.
- [14] A Bellogin, I Cantador, F Dier. "An empirical comparison of social, collaborative filtering, and hybrid recommenders". ACM Transactions on Intelligent Systems and Technology (TIST), 2013, 4(1): 14.
- [15] L Qingwen, X Yan, H Wencho. "Integrating social information into collaborative filtering for celebrities recommendation". Intelligent Information and Database Systems. Springer Berlin Heidelberg, 2013: 109-118.
- [16] M Roth, A Ben-David, D Deutscher. "Suggesting friends using the implicit social graph". Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2010: 233-242.
- [17] I Konstas, V Stathopoulos, J M Jose. "On social networks and collaborative recommendation". Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2009: 195-202.
- [18] T A Dang, E Viennet. "Collaborative filtering in social networks: A community-based approach". Computing, Management and Telecommunications (ComManTel), 2013 International Conference on. IEEE, 2013: 128-133.
- [19] W Yu, G Lin. "Social circle-based algorithm for friend recommendation in online social networks". Chinese Journal of Computers. 2014 (4)
- [20] Y Zhimin, Y Xiangzhan, Z Hongli. "Commodity recommendation algorithm based on social network". Advances in Computer Science and its Applications. Springer Berlin Heidelberg, 2014: 27-33.
- [21] Z Tong, J McAuley, I King. "Leveraging social connections to improve personalized ranking for collaborative filtering". Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management. ACM, 2014: 261-270.
- [22] B Shapira, L Rokach, S Freilikhman. "Facebook single and cross domain data for recommendation systems". User Modeling and User-Adapted Interaction, 2013, 23(2-3): 211-247.
- [23] D Odoherty, S Jouili, R P Van. "Trust-based recommendation: an empirical analysis". Proceedings of the Sixth ACM SIGKDD Workshop on Social Network Mining and Analysis SNA-KDD. Beijing, China, ACM, 2012.
- [24] L Fengkun, H J Lee. "Use of social network information to enhance collaborative filtering performance". Expert systems with applications, 2010, 37(7): 4772-4778.
- [25] R Huigui, H Shengxu, H Chunhua. "User Similarity-based Collaborative Filtering Recommendation Algorithm". Journal on Communications. 2014, (2): 16-24.
- [26] P Cremonesi, Y Koren, R Tumin. "Performance of recommender algorithms on top-n recommendation tasks". Proceedings of the fourth ACM conference on Recommender systems. ACM, 2010: 39-46.
- [27] N Barbieri, G Manco. "An analysis of probabilistic methods for top-N recommendation in collaborative filtering". Machine Learning and Knowledge Discovery in Databases, 2011: 172-187.
- [28] A Bellogin, I Cantador, P Castells. "A comparative study of heterogeneous item recommendations in social systems". Information Sciences, 2013, 221: 142-169.
- [29] X Liang. Recommender Systems in Action. Beijing: Posts & Telecom Press, 2012:35-73.