AOPUT: A Recommendation Framework Based on Social Activities and Content Interests

Yingying Deng*, Tun Lu*, Huanhuan Xia*, Dongsheng Li†, Tiejiang Liu*, Xianghua Ding*, Ning Gu*
*School of Computer Science, Fudan University, Shanghai, China
†Department of Computer Science and Technology, Tongji University, Shanghai, China
dyyp.peace@gmail.com; {lutun, huanhuanxia, liutiejiang, dingx, ninggu}@fudan.edu.cn; dongshengli@tongji.edu.cn

***Abstract*—Content consuming and sharing are two most important user activities in social networking sites (SNSs). Lots of studies have been conducted on content recommendation using users' common interests. However, little has been done to help users to select friends and share content within their social networks. In this paper, we contribute a recommendation framework *AOPUT* to recommend both content and friend list for sharing to users leveraging content and social information in SNSs. It consists of two recommendation components: *Recder* and *ShareAider*. Recder generates content recommendations by connecting users with common interests. An improved Jaccard similarity is proposed to improve the Collaborative Filtering (CF) recommendation quality. ShareAider recommends a friend list to users when they want to share content with their friends. CF method and a social-based method are compared and the combination of them are explored to achieve better results. *AOPUT* is evaluated on a real world social network. The experimental results show that (1) Recder can provide better recommendation quality than the traditional CF method thanks to the improved Jaccard similarity; (2) social-based method performs better than CF since the sharing behavior in SNSs are highly dominated by users' social preferences, and the combination of these two methods performs better than each of them individually.**

***Keywords*—Recommender Systems; Collaborative Filtering; Social Network; Social Activities; Content Interests**

I. Introduction

During the last few years, a large number of social networking sites (SNSs) have emerged. Most of them provide an online service that focuses on building social networks and enhancing friendships among people. These sites are collaborative networks where people meet to exchange experiences, express comments and also share ideas or activities within their individual networks. With the surge of users in SNSs, a large number of topics are posted each day and users' social networks are increasingly large. This leads to two phenomena: (1) if a user was not online for a period of time, the topics she may be interested in are not visible to her in the first few pages when she logs in; (2) when a participant shares a topic, she may face the problem of searching her social network for target friends who are interested in this topic to share with.

Recommender systems (RSs) emerged as a solution to recommend personalized content to users [1]. Collaborative filtering (CF) is a recommendation technique that has been widely used in RSs. It builds a user-item rating matrix with each element denoting the preference from a user for an item [2]. The matrix is utilized to recommend interesting items to a given user. In SNSs, users pay more attention to topics that their familiar friends have participated in and would also like to be a participant to enhance friendships. Users' preferences are influenced by their friends and they influence others as well. However, social relationships among users are ignored in most of the traditional RSs [3].

In the past few years, leveraging trust relationships among users to improve RSs have been explored [4–6]. Most of the work mainly focuses on incorporating the trust relationships into user-item matrix to achieve better results. However, trust relationships are mainly one-way relationships, while social friendships are mutual [3] and there may exist negative relations (distrust) in addition to true friendships (trust) [7]. Moreover, very few sites have implementations of trust mechanisms. Thus recommendations based on trust relationships have substantial limitations when applied to SNSs.

Recently, much work has been done on social recommendations with the development of SNSs [3, 7, 8]. However, a majority of the work focuses on why social information can benefit RSs and how to model such information to improve traditional RSs. Little attention has been paid to generate recommendations by connecting users with common interests and help users to select friends when sharing content within their social networks. The two problems in SNSs are actually identifying the most relevant users given a topic according to users' social activities and content interests. The contributions of this paper are as follows:

- Present a recommendation framework AOPUT, which contains two core tasks: (1) generating content recommendations by connecting users with common interests; (2) recommending a friend list to users when they want to share content with their friends;
- Elaborate that the capture of user preferences is mainly through users' social activities such as commenting or viewing topics, and an improved Jaccard similarity is proposed;
- Demonstrate that the sharing behaviors in SNSs are highly dominated by users' social preferences through a comparison of CF approach and a social-based approach, and the combination of them is explored to achieve better results.

The remainder of this paper is organized as follows: Section 2 provides some related work; Section 3 introduces the AOPUT framework at length; Section 4 details the main algorithms used in AOPUT framework; Results and analysis of experiments are presented in Section 5 and the conclusions and future work are given in Section 6.

II. Related Work

A. Collaborative Recommendation

Traditional RSs are mainly based on CF techniques. In CF, recommendations are generated for a target user based on the relationships between her and a neighborhood of users. Memory-based approaches and model-based approaches are two types of approaches widely studied [9]. Memory-based approaches use the user-item matrix which contains users' preferences on items to generate predictions, while model-based approaches try to train a model leveraging users' ratings [9, 10]. There are two variations of memory-based approaches: (1) user-based approaches, in which similarity between users is computed for all pairs of users, and for a target user, the opinions of her most like-minded users are exploited to make recommendations [11] (2) item-based approaches, in which similarity between items is computed, and given a target user, the most similar items to the ones she has already rated are selected to be the recommendations [11, 12]. The most common "similarity" methods are Pearson correlation coefficient [2] and cosine-based similarity [9], which are widely used in RSs with rating mechanisms.

Memory-based approaches have received considerable attention because of their simplicity and computation efficiency. In our work, user-based approach is utilized to make recommendations. However, the two similarity methods discussed above are not applicable as social activities instead of rating behaviors are the factors indicating users' preferences.

B. Social Recommendation

As introduced in Section 1, a user's preference is influenced by her social relationships. In [13], the authors compared recommendations from friends with online recommenders and found that trusted friends' recommendations are preferred. Recently, much work has been done to improve RSs by leveraging social factors. [7] pointed out the fact that most social networks only contain acquaintance relationships, without distinguishing trust from distrust. The authors proposed unsupervised and semi-supervised algorithms to distinguish the two relationships to get a trust-distrust social network graph, which can be more informative for behavior prediction. [8] evaluated a Random Walk with Restarts (RWR) model on Last.fm dataset and showed that the model system performs better when incorporating the information of friendship and social tagging and outperforms the standard CF approach. [3] proposed a matrix factorization framework with social regularization, which is used for improving RSs by incorporating social network information.

Different from these social RSs which attempt to model for social information such as friend networks and social tags to improve traditional RSs, our work focuses on finding the most relevant users given a topic according to its dynamically changed context information such as its participants.

III. Overview of AOPUT

As mentioned earlier, AOPUT mainly has two tasks: (1) generating content recommendations by connecting users with common interests; (2) recommending a friend list to users when they want to share content with their friends. The two tasks are mainly completed by Recder and ShareAider respectively, which are two core components in AOPUT. Considering a candidate target user, her relevance score for a topic is computed by Recder utilizing the social activities of her friends and her own. Another component, ShareAider, is responsible for providing a convenience for sharers by ranking the potential friends to be shared with in front of the recommended friend list.

Fig. 1 gives the overview of AOPUT. In addition to the core components of Recder and ShareAider, there are three other components: Event Capture, Sender and Updater. Event Capture is responsible for capturing users' actions and delivering them to Recder and ShareAider according to their activity types. Sender is responsible for sending the recommended topics to target users via online notification. Another component Updater, which is called by Recder to save the recommended topics in PITable, also captures common users' activities to update PITable, such as marking a topic as "participated" and removing it. Considering a topic, which is the active topic, and a participant or a sharer corresponding to it, the overall working process after capturing a participation or a share action and how the components collaborate with each other are outlined as below.

1) ***Recommending a topic to users:*** When a user joins in a topic, Recder generates target users who are the most potentially relevant ones towards the topic according to its existing participants based on CF approach. Recder follows the five steps to recommend the topic to potential users after receiving a new activity from Event Capture: (1) select the new participant's friends as the original candidates; (2) determine whether the participant has joined in this topic before, and if so, extract her friends that have been recommended this topic to and remove them from the candidates; (3) compute the relevance scores of the candidate set for the topic according to their social activitie history; (4) identify the potential top-N target users who have the first N highest relevance scores and finally, (5) call Sender to deliver the topic to the N users and call Updater to save the topic in PITable.

Recder considers a novel recommendation mechanism that a topic can be recommended to a user when one of her friends has participated in it. The participation of friends can provide a good explanation of recommendations [14]. Moreover, the topics that a user's friends have participated in may also be those this user would not like to miss.

AOPUT maintains a table, PITable, which stores the most relevant top-k topics for each user. Given a user, the topics are those k most relevant ones she has not seen or joined in and they are recommended to the user by Sender via notification when she logs in the site. Specifically, when a new recommendation is generated to an online user, it will be sent to the user immediately via online notification with the consideration of real-time participation. Furthermore, Updater guarantees that there are at most k topics for each user in PITable. An update operation will be triggered if there is a new recommended topic and topics that the corresponding relevant users have just joined in will be removed.

2) ***Recommending a friend list to a sharer:*** When a participant wants to share a topic with one or some of her friends, she may need to search her social network for target

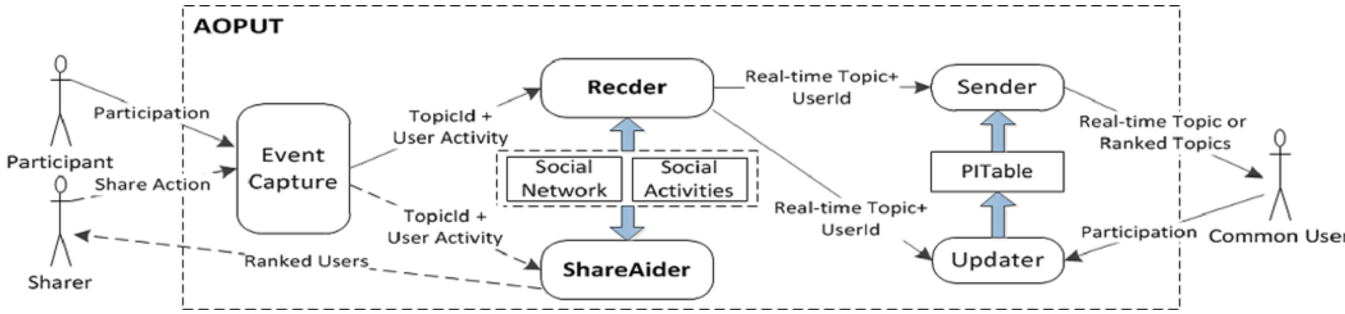

Fig. 1. Components of AOPUT

users. ShareAider recommends a friend list to aid the sharer to select friends. In our work, two mechanisms are considered to generate candidates. One is based on CF approach, which is the same as the approach Recder uses. The other mechanism is based on a social-based, non-CF approach. It is not related with the topic and depends on the sharer's share history to generate the friend list. The two approaches are compared in ShareAider and their recommendation results are combined using a linear model to achieve better performance.

IV. Algorithms

In this section, We introduce the CF approach used in Recder and ShareAider and the social-based approach used in ShareAider. The algorithms involved in these approaches are also presented.

A. The CF Approach

In our work, memory-based CF approach is utilized to make recommendations. As AOPUT finds the most relevant users given a topic according to its existing participants, the user-based CF approach is more appropriate than the item-based CF approach. In the process of recommending a topic to users, the third and the fourth steps are actually parts of work of a standard CF approach. Based on [15], the user-based approach can be separated into the following three steps:

1) For each user u in the candidate set, compute the similarity between her and each of the existing participants;
2) Select m most similar participants to u and compute the relevance score of u for the active topic based on these participants;
3) Reorder all the users in the candidate set based on their relevance scores and identify the top-N target users.

1) ***Improved Similarity Computation****:* The similarity computation in step 1) is a critical step [11]. Pearson correlation coefficient [2] and cosine-based similarity [9] are widely used to compute similarity in systems with rating mechanisms, especially with explicit ratings. However, very few SNSs have implementations of explicit rating mechanisms. In most SNSs, a user's interest can be captured by her social activities such as viewing or commenting topics. Based on the intuition that different types of users' social activities represent their dissimilar interests and comment behaviors indicate higher interests than view behaviors, we assign higher relevance scores for a topic to users with comment behaviors than those with view behaviors. In this paper, the behaviors that a user shares a topic or is shared with on a topic are regarded as the comment behavior, which denotes a higher relevance for the topic than view behavior.

The high similarity between two users is determined by their highly similar rating behaviors using Pearson correlation coefficient or cosine-based similarity. The same similarity value between two users can be achieved in such situations: (1) user A and user B view a same topic and (2) user A and user C comment another topic. In fact, the similarity between A and B is lower than that between A and C as the view behavior indicates a lower relevance of a user towards a topic than comment behavior. It is reasonable to assume that two users both having a higher relevance towards a topic have higher similarity themselves. Thus the similarity between A and C should be different from that between A and B.

In some researches on content recommendation [16, 17], Jaccard coefficient is used to compute the similarity between users or items. [16] described the recommendations for users of Google News, where a user's preference is inferred by which news she has clicked. [17] proposed a system for recommendation in online social communities with the assumption that a user's click/view on a content item indicates her interest in it. The Jaccard similarity measure takes values between 0 and 1, where 1 denotes liked or viewed and 0 denotes disliked or not viewed yet. Similar to that, we use 1 to denote participated and 0 to denote not participated yet. Thus the similarity between two users u_i, u_j can be defined as:

$$sim(u_i,u_j) = \frac{|T_{u_i} \cap T_{u_j}|}{|T_{u_i} \cup T_{u_j}|} \tag{1}$$

where T_{u_i} is the set of topics u_i has participated in, and T_{u_j} is the set of topics u_j has joined in. For convenience, we call this method "Naive Jaccard".

[18] extended the standard Jaccard to estimate user similarity with the consideration of statistical fluctuations. The fluctuation is: user pairs with a small sum of topics each of them has joined in are likely to achieve higher similarity. The extended Jaccard similarity measure is:

$$sim(u_i,u_j) = \frac{|T_{u_i} \cap T_{u_j}|}{|T_{u_i} \cup T_{u_j}|}(1 - \frac{\Theta}{|T_{u_i} \cup T_{u_j}|}) \tag{2}$$

where Θ is a factor determining how strongly the user pairs with few participated topics should be penalized. The authors found that it can yield optimal results with Θ being set to a certain value. We call this method "Medo Jaccard".

As has been analyzed before, the behaviors of viewing and commenting a topic should be treated differently. Thus

(1) and (2) will not be applicable in our tests. We explore a new similarity computation method, "Extended Jaccard", which also revises the standard Jaccard. It uses the following formula to compute the similarity between users:

$$sim(u_i,u_j) = \frac{|T_{u_i} \cap T_{u_j}| + (\Delta - 1)T^c}{|T_{u_i} \cup T_{u_j}| + (\Delta - 1)T^c}\left(1 - \frac{\Theta}{|T_{u_i} \cup T_{u_j}| + (\Delta - 1)T^c}\right) \tag{3}$$

where T^c is the number of topics that u_i and u_j have commented commonly and Δ denotes the weight we assign to comment behavior with the assumption that the weight assigned to view behavior is 1. In our tests, Δ = 4 and Θ = 0.7 seemed to perform the best.

*2) **Selection of Target Users**:* The relevance score of each candidate u_c towards a topic is the sum of similarities between u_c and the existing participants of this topic:

$$rscore(u_c,t) = \sum_{p \in P_t} sim(u_c,p) \tag{4}$$

where t is a specific topic and P_t is the set of existing participants of t. All candidate users are sorted by their relevance scores towards t and the top-N users with highest scores will be selected as the target users to recommend to.

In SNSs, two users who are not friends may have few topics that jointly participated in as the topic sets that are visible to them may be different. Considering the fact that a topic may have lots of participants who have low similarity with u_c, we select m most similar participants to u_c to sum over to reduce the effect of those participants with a fairly low similarity with u_c. In our tests, setting m = 5 seemed to perform the best.

B. The Social-based Approach

The social-based approach recommends candidates for a sharer based on her share history. In contrast to the CF approach using a topic's context participants to find target users, the social-based just considers the candidates' social relationships with the sharer, which is irrelevant to the topic.

In this paper, the sharing type considered by AOPUT is "At" (@). At is a very convenient method for users to communicate with each other. It has been adopted by many famous SNSs such as Twitter (*https://twitter.com*), Renren (*http://www.renren.com*), Weibo (*http://weibo.com*). User A can At user B through inputting "@" with B's id or name followed when publishing or commenting a topic. Then there will be an At notification for B, which contains the information of the topic. Renren and Weibo, two Chinese SNSs, have implemented to pop up a friend list for A according to her share history when she inputs "@". Based on this idea, our social-based approach to define the probability that a candidate user u_c will be selected by a sharer s is:

$$bselect(u_c,s) = \sum_{i}^{SN} \frac{1}{\sqrt{d_i}} \tag{5}$$

where SN denotes the times that s has shared topics with u_c and d_i is the interval days between when the ith time a topic is shared and the current time this function is called.

C. Combination of CF and Social-based Approach

In this paper, we also consider the combination of CF and social-based approach. We use the combination to generate a friend list for a sharer as the social-based approach is just called when there is a share action. The combination of the two approaches is defined as the combination of the friend lists generated by the two approaches through a linear model:

$$comL(L_c,L_s) = \sum_{i}^{N} xL_{ci} + \sum_{i}^{N} yL_{si} \tag{6}$$

where L_c and L_s are the friend lists generated by CF and social-based approach respectively. N is the size of the recommended list, and L_{ci} denotes the ith friend in L_c, while L_{si} is the ith friend in L_s. x and y are the proportion of the two lists and the sum of them is 1. The friends in the result list is sorted in descending order according to the relevance score and the top-N friends are extracted as the target users. In our tests, the best results can be achieved when x and y are assigned with approximate equal weights.

V. EXPERIMENTS AND EVALUATIONS

In this section, we conduct several experiments using AceBridge (*http://www.acebridge.net*) dataset to evaluate the novel recommendation mechanism and the different approaches described earlier.

A. Dataset

AceBridge, a global elite community, is a typical SNS. There are more than 6,400 registered users in AceBridge to date. The data we collected from this site consists of users' friend network and social activities for 60 days. The friend network contains the information of friend relationships between users. It is used to identify the candidate user set when a new user joins in a topic. The social activity data records users' activities during the collection period. Each record contains the meta data of a topic, information of the user that joins in this topic, user's activity type and participation time. The activity data is divided into (1) a training set, which is used to compute the similarity between users and (2) a testing set, on which we evaluate the recommendation mechanism and different approaches described in previous section. Each user in the site has a popularity attribute, the high value of which indicates the corresponding user is "active". A user's popularity is determined by factors such as the number of topics she has published and commented, the number of times she has visited the site. In our tests, if the size of the recommendation list is smaller than N, users with high popularity values will be leveraged to complete the list.

B. Coverage of Recder Recommendation

In the experiments, Naive Jaccard, Medo Jaccard and Extended Jaccard are utilized to compute the similarity between users based on the training set respectively. The corresponding similarity results are used to generate user recommendation lists in testing set. Naive, Medo and Extended are three CF approaches corresponding to the above similarity methods. We also use a benchmark to evaluate against. The benchmark is a non-CF approach to identify target users according to their popularity. It is natural and we believe it is highly appropriate

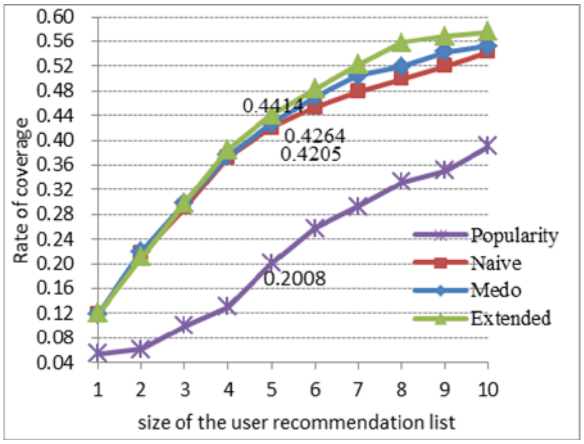

Fig. 2. Recommendation results of Recder

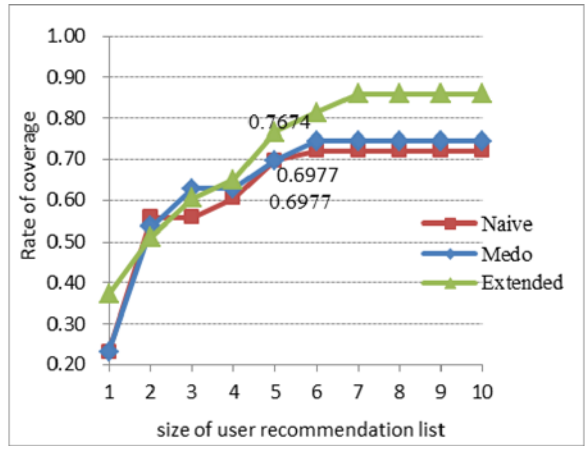

Fig. 3. Three CF approaches evaluated on share data

as users who are more active are more likely to join in more topics and be shared with by sharers. In our tests, the size N of the user recommendation list ranges from [1, 10], and if needed, active users will be used to complete the list.

Recder works by computing the most relevant users towards a topic when a new participant joins in it. Thus we compare the recommended users with the real related users to the topic to evaluate Recder.

Fig. 2 shows the proportion of four kinds of recommended users in the real related users, or rather the coverage rate of the recommendation results of the four approaches. The real related users are those who belong to the new participant's friend network and also participated in the topic after the new participant joined in. They can be viewers, commenters, sharers or users who are shared with by sharers. Each point on the curves in Fig. 2 denotes the coverage rate when N is the corresponding value of the abscissa axis. From the figure, we can see that the CF approaches are significantly better than Popularity approach. Specifically, the coverage rate of Extended is 44.14% when N = 5, while 5 only accounts for 2.5% of average number of friends of all the new participants tested, i.e. the coverage rate is just 2.5% if using a random method to generate the target users. The figure also shows that the Extended approach based on our improved similarity method performs better than Naive and Medo approaches when N ranges from [4, 10] and has the identical performance when N is 1 to 3.

To evaluate the three CF approaches further, we compare them when recommending friend lists to sharers. In this case, the real related users are those participants who are actually shared with by sharers. Fig. 3 shows the coverage rate of Naive, Medo and Extended when evaluated on the share data extracted from the testing set. We can see that the Extended approach achieves better performance than the other two approaches on the whole. It is noteworthy that Extended has a 14 percent higher coverage when N = 1, which indicates that it can locate more relevant users at the first position of the list.

C. CF Approach vs. Social-based Approach

As introduced earlier in this paper, both CF approach and social-based approach can be used to identify the target friends to share with for a sharer. The recommended friends are compared with users that are actually shared with by sharers. Evaluation of the two approaches is similar to the evaluation of the three CF approaches as Fig. 3 shows. We use Extended CF approach to represent CF approaches. Fig. 4 shows the performance of CF approach, social-based approach and the combination of the two approaches. It is evident that the social-based approach performs much better than CF approach, which indicates that the sharing behaviors of users in SNSs are highly affected by the social relationships among users and have less dependence on the interest the target users have in the active topic. Sharers are more inclined to share topics with friends they are familiar with or they have frequent interactions with. In the tests, we also combine the two approaches using a linear model described in Section 4. The results of the combination are best when the recommended friend lists of the two approaches are assigned with approximate equal weights. From Fig. 4, we can also figure out that the combination performs better than each of the two approaches.

D. Discussion

The Extended Jaccard similarity we proposed is an improved method based on Jaccard similarity measure. It increases the weight of two activities if they are both comment behaviors on a topic. In SNSs, the topics are generally events, ideas and other kinds of content that small range of users participate in. Thus two users commenting the same topic indicates high similarity between them. We can see from Fig. 2 that Extended performs better than Naive and Medo, but the improvement is not obvious. By analysing the dataset, we found that the sum of topics have been commented commonly by all user pairs only accounts for 1.5% of the sum of topics that have been commonly participated in. Such small proportion weakens the effect of weights increasing mechanism.

However, Extended shows much better results in Fig. 3. It is mainly due to the different dataset we evaluated on. In Fig. 3, the evaluation is based on the share data extracted from the testing set. The real related users to compare to are those who are actually shared with by sharers, while in the evaluation as Fig. 2 shows, the users also include viewers, commenters and sharers. The number of viewers accounts for a largest proportion. As mentioned earlier, the behavior that a user shares a topic or is shared with on a topic is regarded as

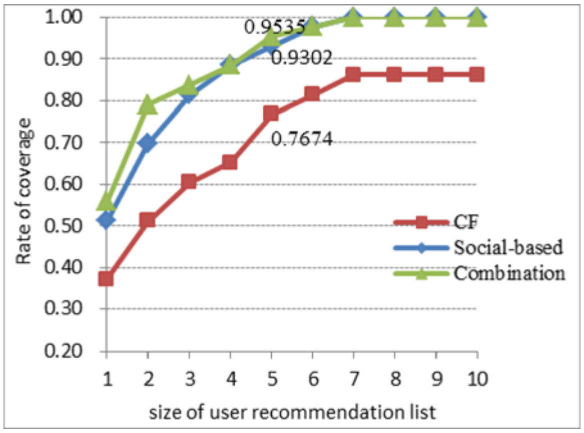

Fig. 4. Performance of CF approach, social-based approach and the combination of the two approaches evaluated on share data

the comment behavior. Thus for a topic, users who are selected by a sharer to share this topic with are more relevant than those who just view the topic. Different from Naive Jaccard and Medo Jaccard, Extended Jaccard distinguishes the different relevances, which can achieve much better results.

Fig. 4 shows that CF approach performs not very well compared to social-based approach when used to predict the target friends for sharers to share with. The curve of social-based approach in Fig. 4 is much closer to the curve of the combination than that of CF approach. From this, we can infer that sharing behaviors in SNSs are closely related with the social interactions between users even though the users selected by a sharer are not so relevant towards the corresponding topic. The traditional CF approaches may need to change much to adapt to social recommendations especially in the situation where social interactions are very important.

VI. Conclusions and Future Work

This paper presents a recommendation framework *AOPUT* that can be used in SNSs. Users collaborate through participating in topics. Each user's activities affect recommendations generated to other users. AOPUT adopts a novel recommendation mechanism, under which a user is recommended with topics only when one of her friends has joined in it. This mechanism provides a good explanation of recommendations due to its consideration of friends' participation [14]. In addition, we propose an improved Jaccard similarity to improve the CF approach. Our experimental results on a real life dataset show that it performs better than traditional CF approach. AOPUT also aids sharers when selecting friends to share topics with. A social-based approach is proposed and is compared with the CF approach. The experimental results indicate that sharing behaviors in SNSs are closely related with the social interactions between users, and the combination of the two approaches is explored to achieve better results. In the future, we plan to investigate how to incorporate the frequency that a user participates in a topic into the similarity methods to further improve our work.

Acknowledgment

The work is supported by the National Natural Science Foundation of China (NSFC) under Grants No.61272533 & No.61233016, and Shanghai Science & Technology Committee Key Fundamental Research Project under Grant No.11JC1400800 & No.12dz1508600, and ChinaGrid Project.

References

[1] P. Resnick and H. R. Varian, "Recommender systems," *Commun. ACM*, vol. 40, no. 3, pp. 56–58, 1997.
[2] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *CSCW*, 1994, pp. 175–186.
[3] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *WSDM*, 2011, pp. 287–296.
[4] P. Massa and P. Avesani, "Trust-aware collaborative filtering for recommender systems," in *CoopIS/DOA/ODBASE (1)*, 2004, pp. 492–508.
[5] L. Zanette, C. L. R. da Motta, F. M. Santoro, and M. Elia, "A trust-based recommender system for collaborative networks," in *CSCWD*, 2009, pp. 197–202.
[6] J. O'Donovan and B. Smyth, "Trust in recommender systems," in *IUI*, 2005, pp. 167–174.
[7] S.-H. Yang, A. J. Smola, B. Long, H. Zha, and Y. Chang, "Friend or frenemy?: predicting signed ties in social networks," in *SIGIR*, 2012, pp. 555–564.
[8] I. Konstas, V. Stathopoulos, and J. M. Jose, "On social networks and collaborative recommendation," in *SIGIR*, 2009, pp. 195–202.
[9] J. S. Breese, D. Heckerman, and C. M. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *UAI*, 1998, pp. 43–52.
[10] Y. Koren, R. M. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Computer*, vol. 42, no. 8, pp. 30–37, 2009.
[11] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artificial Intellegence*, vol. 2009, 2009.
[12] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *WWW*, 2001, pp. 285–295.
[13] R. Sinha and K. Swearingen, "Comparing recommendations made by online systems and friends," in *Proceedings of the DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*, 2001.
[14] G. Groh, S. Birnkammerer, and V. Kllhofer, "Social recommender systems," in *Recommender Systems for the Social Web*, ser. Intelligent Systems Reference Library, 2012, vol. 32, pp. 3–42.
[15] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering," in *SIGIR*, 1999, pp. 230–237.
[16] A. Das, M. Datar, A. Garg, and S. Rajaram, "Google news personalization: scalable online collaborative filtering," in *WWW*, 2007, pp. 271–280.
[17] D. Li, Q. Lv, X. Xie, L. Shang, H. Xia, T. Lu, and N. Gu, "Interest-based real-time content recommendation in online social communities," *Knowledge-Based Systems*, vol. 28, no. 0, pp. 1 – 12, 2012.
[18] M. Medo, Y.-C. Zhang, and T. Zhou, "Adaptive model for recommendation of news," *CoRR*, vol. abs/0910.3490, 2009.