Service Recommendation Based on Social Balance Theory and Collaborative Filtering

Lianyong Qi[1,2,3(✉)], Wanchun Dou[2], and Xuyun Zhang[2,4]

[1] State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072, China
lianyongqi@gmail.com
[2] State Key Laboratory for Novel Software Technology, Department of Computer Science and Technology, Nanjing University, Nanjing 210023, China
douwc@nju.edu.cn, xuyun.zhang@auckland.ac.nz
[3] School of Information Science and Engineering, Qufu Normal University, Rizhao 276826, China
[4] Department of Electrical and Computer Engineering, University of Auckland, Auckland, New Zealand

Abstract. With the increasing popularity of web service technology, many users turn to look for appropriate web services to further build their complex business applications. As an effective manner for service discovery, service recommendation technique is gaining ever-increasing attention, e.g., Collaborative Filtering (i.e., CF) recommendation. Generally, the traditional CF recommendation (e.g., user-based CF, item-based CF or hybrid CF) can achieve good recommendation results. However, due to the inherent sparsity of user-service rating data, it is possible that the target user has no similar friends and the services preferred by target user own no similar services. In this exceptional situation, traditional CF recommendation approaches cannot deliver an accurate recommendation result. In view of this shortcoming, a novel Social Balance Theory (i.e., SBT)-based service recommendation approach, i.e., Rec_{SBT} is introduced in this paper, to help improve the recommendation performance. Finally, through a set of simulation experiments deployed on MovieLens-1M dataset, we further validate the feasibility of Rec_{SBT} in terms of recommendation accuracy and recall.

Keywords: Service recommendation · Target user · Friend user · Enemy user · Social balance theory · Collaborative filtering

1 Introduction

With the gradual popularity of SOA (Service Oriented Architecture), the available web service number in service communities is becoming increasingly larger. In this situation, many users are apt to find their interested web services through various recommendation techniques, e.g., well-known Collaborative Filtering (i.e., CF; e.g., user-based CF, item-based CF or hybrid CF) [1]. In CF recommendation, through analyzing known user-service rating data (only the subjective rating data is considered in this paper),

Q.Z. Sheng et al. (Eds.): ICSOC 2016, LNCS 9936, pp. 637–645, 2016.
DOI: 10.1007/978-3-319-46295-0_43

we can first determine the target user's similar friends or the target service (i.e., the service preferred by target user)'s similar services, and further recommend appropriate services to the target user.

While due to the inherent sparsity of user-service rating data [2], in certain situations, the target user does not have any similar friend and the target services do not own any similar service. In this situation, traditional CF recommendation approaches cannot deliver an accurate recommendation result, which brings a big challenge for recommendation effect. In view of this challenge, a novel Social Balance Theory [3] (i.e., SBT)-based service recommendation approach, i.e., Rec_{SBT} (Recommendation based on SBT) is put forward in this paper, to help improve the recommendation performance. Different from the traditional CF recommendation approaches, in Rec_{SBT}, we first look for the target user's "enemy" (i.e., antonym of "friend"), and further determine the target user's "possible friends" based on Social Balance Theory (e.g., "enemy's enemy is a friend" rule, "friend's enemy is an enemy" rule, "enemy's friend is an enemy" rule); finally, the services preferred by target user's "possible friends" are recommended to the target user.

The rest of paper is structured as below. In Sect. 2, we formalize the service recommendation problem and clarify the paper motivation. A novel service recommendation approach, i.e., Rec_{SBT} is brought forth in Sect. 3. In Sect. 4, we design a set of experiments to validate the feasibility of Rec_{SBT}. In Sect. 5, we introduce the related works and compare them with our approach. Finally, conclusions are presented in Sect. 6.

2 Formal Specification and Motivation

2.1 Formal Specification

Generally, the service recommendation problem could be formalized with following *Web_Ser_Rec* (*User_set, WS_set, Rating_set*, $user_{target}$), where

(1) *User_set* = $\{user_1, \ldots, user_m\}$: $user_i$ $(1 \le i \le m)$ denotes a user in web service community and m is the number of users.
(2) *WS_set* = $\{ws_1, \ldots, ws_n\}$: ws_j $(1 \le j \le n)$ denotes a service in web service community and n is the number of web services.
(3) *Rating_set* = $\{r_{i-j} \mid 1 \le i \le m, 1 \le j \le n\}$: r_{i-j} denotes $user_i$'s rating over service ws_j. As our previous work [4] did, the popular 1* $\sim$ 5* rating scores are adopted here to depict r_{i-j}.
(4) $user_{target}$: target user who requires service recommendation, and $user_{target} \in$ *User_set* holds here.

With the formal specification, we can clarify the service recommendation problem as below: according to the known user-service rating data (in *Rating_set*) between users (in *User_set*) and services (in *WS_set*), recommend appropriate services from *WS_set* to the target user $user_{target}$.

2.2 Motivation

Next, we demonstrate the motivation of our paper with the example presented in Fig. 1. In the example, user set $User\_set$ = {$John$, $Lily$, $Jack$} ($user_{target}$ is $John$) and service sets WS_set = {ws_1, …, ws_6}. The user-service rating data (i.e., $Rating\_set$) is also shown in Fig. 1. As Fig. 1 shows, target user $John$ prefers services ws_1 and ws_2; therefore, ws_1 and ws_2 are called "target services" in the rest of paper.

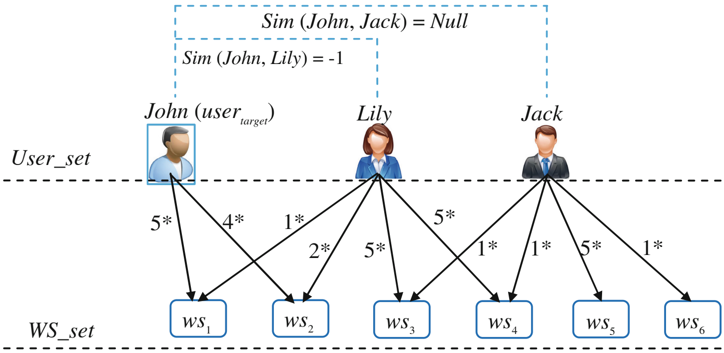

Fig. 1. Service recommendation scenario: an example

With the known data of $User\_set$, WS_set and $Rating\_set$, we can calculate the similarity between different users by the well-known Pearson Correlation Coefficient (i.e., PCC) [5]. Concretely, similarity Sim ($John$, $Lily$) = −0.27 and Sim ($John$, $Jack$) = $Null$ (as $John$ and $Jack$ have not invoked common web services). Likewise, we can also calculate the similarity between target services (i.e., ws_1 and ws_2) and other services (i.e., ws_3, ws_4, ws_5, ws_6). Concretely, Sim (ws_1, ws_3) = Sim (ws_1, ws_4) = Sim (ws_2, ws_3) = Sim (ws_2, ws_4) = −1, while Sim (ws_1, ws_5) = Sim (ws_1, ws_6) = Sim (ws_2, ws_5) = Sim (ws_2, ws_6) = $Null$.

With the above calculation, a conclusion could be drawn that the target user (i.e., $John$) has no similar friends and the target services (i.e., ws_1 and ws_2) own no similar services. In this situation, traditional CF recommendation approaches cannot deliver an accurate recommendation result. In view of this shortcoming, we introduce Social Balance Theory into service recommendation and bring forth a novel recommendation approach Rec_{SBT} in the next section.

3 SBT-Based Service Recommendation

3.1 Social Balance Theory

Social Balance Theory analyzes and formalizes the social relationships among involved three parties, and provides us a new perspective for friend recommendation in social network. Concretely, there are several intuitive rules in Social Balance Theory, e.g., "enemy's enemy is a friend", "enemy's friend is an enemy" and "friend's enemy is an enemy" (The details of SBT are omitted here due to page limit. Readers who are interested in SBT can turn to work [3] for reference).

3.2 *Rec*Rec_{SBT}: A Service Recommendation Approach

Next, we introduce a novel service recommendation approach Rec_{SBT} by considering the rules in Social Balance Theory. Concretely, our proposal consists of the following three steps (see Fig. 2).

Step1: Determine "friend" or "enemy" relationship between different users. For different $user_i$ and $user_j$ $(i \neq j)$, calculate their similarity $Sim(user_i, user_j)$ and further determine the "friend" or "enemy" relationship between them.

Step2: Determine "possible friends" of target user based on SBT. For target user $user_{target}$, determine his/her "possible friend" set *Possible_friend_set* ($user_{target}$) based on "enemy's enemy is a friend" rule, "enemy's friend is an enemy" rule and "friend's enemy is an enemy" rule in Social Balance Theory.

Step3: Service recommendation. For the "possible friends" in set *Possible_friend_set* ($user_{target}$), determine their preferred services and recommend them to $user_{target}$.

Fig. 2. Three steps of our proposed recommendation approach Rec_{SBT}

(1) **Step1: Determine "friend" or "enemy" relationship between different users.**

First, for two different users $user_i$ and $user_j$ ($user_i$, $user_j \in User\_set$ and $i \neq j$), we can calculate their similarity $Sim(user_i, user_j)$ based on PCC technique, whose formula is specified in (1). Here, set *Common_ser_set* denotes the common service set that have been invoked and rated by $user_i$ and $user_j$; r_{i-k} and r_{j-k} denote web service ws_k's rating scores by $user_i$ and $user_j$ respectively; $\overline{r_i}$ and $\overline{r_j}$ represent $user_i$'s and $user_j$'s average rating scores over all his invoked services. Specially, if $user_i$ and $user_j$ have not invoked same services before (i.e., *Common_ser_set* = *Null*), then $Sim(user_i, user_j)$ = *Null* holds.

$$Sim(user_i, user_j) = \frac{\sum_{ws_k \in Common\_ser\_set} (r_{i-k} - \overline{r_i}) * (r_{j-k} - \overline{r_j})}{\sqrt{\sum_{ws_k \in Common\_ser\_set} (r_{i-k} - \overline{r_i})^2} * \sqrt{\sum_{ws_k \in Common\_ser\_set} (r_{j-k} - \overline{r_j})^2}} \tag{1}$$

Afterwards, according to the derived similarity $Sim(user_i, user_j)$ in (1), we can further determine the "friend" or "enemy" relationship between $user_i$ and $user_j$ by (2). In (2), Q $(0.5 \leq Q \leq 1)$ is a pre-set similarity threshold for "friend" relationship; correspondingly, $-Q$ $(-1 \leq -Q \leq -0.5)$ is a pre-set similarity threshold for "enemy" relationship. While $Friend\_set(user_i)$ and $Enemy\_set(user_i)$ denote friend set and enemy set of $user_i$, respectively.

$$user_j \begin{cases} \in Friend\_set(user_i) \text{ if } Sim\left(user_i, user_j\right) \geq Q \\ \in Enemy\_set(user_i) \text{ if } Sim\left(user_i, user_j\right) \leq -Q \end{cases} \tag{2}$$

(2) **Step2: Determine "possible friends" of target user based on SBT.**

As Fig. 1 shows, our paper only focuses on the service recommendation scenario where target user has no similar friends; so *Friend_set*($user_{target}$) = *Null* holds based on (2). Namely, we can (but not definitely) only obtain the enemy set *Enemy_set* ($user_{target}$) of target user. Next, we introduce how to get the "possible friends" of target user, based on the derived set *Enemy_set*($user_{target}$) (in Step1) and Social Balance Theory. Concretely, Step2 consists of the following two substeps (see Fig. 3).

Substep2.1: For each $user_x \in Enemy\_set(user_{target})$, determine his enemy $user_y$ (i.e., $user_y \in Enemy\_set(user_x)$) based on (1) and (2). Then according to "enemy's enemy is a friend" rule in SBT, we can conclude that $user_y$ is a candidate "possible friend" of $user_{target}$ and the credibility could be measured by *Friend_probability* ($user_{target}$, $user_y$) in (3). Afterwards, $user_y$ is regarded as a qualified "possible friend" (denoted by set *Possible_friend_set* ($user_{target}$)) of $user_{target}$, if condition in (4) holds (here, Q denotes the pre-set user similarity threshold in (2)).

$$Friend\_probability\left(user_{target}, user_y\right) = Sim\left(user_{target}, user_x\right) * Sim\left(user_x, user_y\right) \tag{3}$$

$$Friend\_probability\left(user_{target}, user_y\right) \geq Q \tag{4}$$

Substep2.2: For each $user_x \in Enemy\_set(user_{target})$, determine his friend $user_z$ (i.e., $user_z \in Friend\_set(user_x)$) based on (1) and (2). Then according to "enemy's friend is an enemy" rule in SBT, we can also infer that $user_z$ is a candidate "enemy" of $user_{target}$ and the credibility could be calculated by *Enemy_probability* ($user_{target}$, $user_z$) in (5). Afterwards, if condition in (6) holds, $user_z$ is considered as a qualified "enemy" of $user_{target}$ and put in set *Enemy_set*($user_{target}$). Similarly, for each $user_y \in Possible\_friend\_set$ ($user_{target}$), determine his enemy $user_k$ (i.e., $user_k \in Enemy\_set(user_y)$) based on (1)–(2). Then according to "friend's enemy is an enemy" rule in SBT, we can infer that $user_k$ is a candidate "enemy" of $user_{target}$ and the credibility could be calculated by *Enemy_probability* ($user_{target}$, $user_k$) in (7). Afterwards, if condition (8) holds, $user_k$ is regarded as a qualified "enemy" of $user_{target}$ and put in *Enemy_set* ($user_{target}$).

$$Enemy\_probability\left(user_{target}, user_z\right) = Sim\left(user_{target}, user_x\right) * Sim\left(user_x, user_z\right) \tag{5}$$

$$Enemy\_probability\left(user_{target}, user_z\right) \leq -Q \tag{6}$$

$$\begin{aligned} Enemy\_probability\left(user_{target}, user_k\right) = {} & Friend\_probability\left(user_{target}, user_y\right) \\ & * Sim\left(user_y, user_k\right) \end{aligned} \tag{7}$$

$$Enemy\_probability\left(user_{target}, user_k\right) \leq -Q \tag{8}$$

Repeat Step1, Substep2.1 and Substep2.2 until the "possible friend" set of target user, i.e., *Possible_friend_set* ($user_{target}$) stays stable. Then we can obtain the target user's possible friends, i.e., $user_y \in$ *Possible_friend_set* ($user_{target}$).

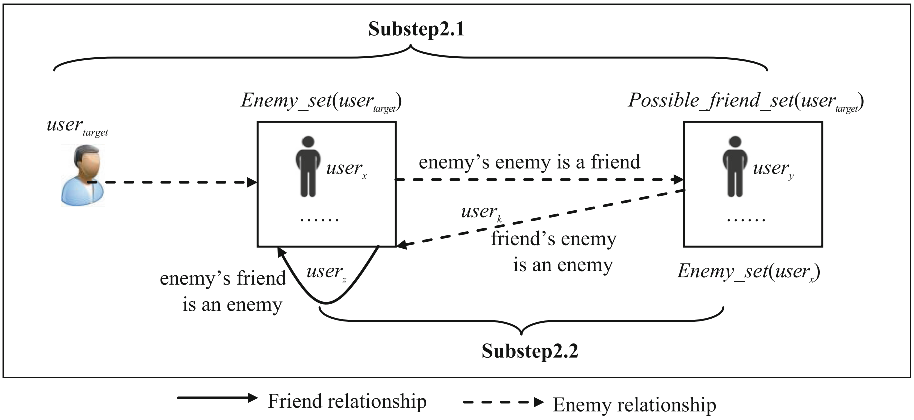

Fig. 3. Relationships of $user_{target}$, $user_x$, $user_y$, $user_z$ and $user_k$ in Step2

(3) **Step3: Service recommendation.**

After Step1 and Step2, we can obtain the target user's "possible friend" set *Possible_friend_set* ($user_{target}$). Next, for each $user_y \in$ *Possible_friend_set* ($user_{target}$), we select his/her preferred services (e.g., with 4* or 5* rating from $user_y$) and recommend them to the target user, so as to finish the whole service recommendation process.

4 Experiment Analyses

4.1 Experiment Dataset and Deployment

Our paper aims at the service recommendation problem with subjective user-service rating data. However, the available service rating data is really rare. Therefore, the popular MovieLens-1M [6] dataset is adopted here for simulation. MovieLens-1M contains 1000209 user-movie ratings from 6040 users over 3952 movies.

In our experiments, the service recommendation accuracy (i.e., *MAE*) and recall are tested respectively (due to the page limit, detailed calculation formula of accuracy and recall is omitted here). Besides, our proposed Rec_{SBT} approach is compared with another two ones, i.e., *WSRec* [7] and *SBT-SR* [4]. The experiments are deployed on a Lenovo PC (2.40 GHz CPU, 2.0 GB RAM), and the software configuration environment is: Windows 7 + JAVA 1.5.

4.2 Experiment Results

In our experiments, m is the number of users and Q denotes the user similarity threshold defined in Eq. (2). In the following two experiment profiles, m is varied from 200 to 1000 and $Q = 0.5$ holds.

(1) ***Profile*1: Service recommendation accuracy**

The *MAE* values of different approaches (i.e., *WSRec*, *SBT-SR* and Rec_{SBT}) are tested respectively, and their execution results are presented in Fig. 4. As shown in Fig. 4, *WSRec*'s recommendation accuracy is low (i.e., *MAE* is high), as only the average rating of the services invoked by target user is considered in *WSRec*. While Rec_{SBT} and *SBT-SR* achieve better accuracy than *WSRec*, which is because more social relationship information among different users are considered. Besides, the recommendation accuracy of Rec_{SBT} and *SBT-SR* are approximate, because the two approaches both consider the "enemy's enemy is a friend" rule in SBT.

(2) ***Profile*2: Service recommendation recall**

The recommendation recall values of three approaches are tested and presented in Fig. 5. It can be seen from Fig. 5 that *WSRec*'s recommendation recall is low, as the average idea is adopted in *WSRec*. Besides, the recommendation recall values of Rec_{SBT} and *SBT-SR* both increase with the growth of m, this is because more "possible friends" of target user could be found when the user number increases. Furthermore, our proposed Rec_{SBT} outperforms *SBT-SR* in terms of recommendation recall, which is due to the fact that *SBT-SR* considers "enemy's enemy is a friend" rule only, while our proposal considers "enemy's enemy is a friend" rule, "friend's enemy is an enemy" rule and "enemy's friend is an enemy" rule simultaneously.

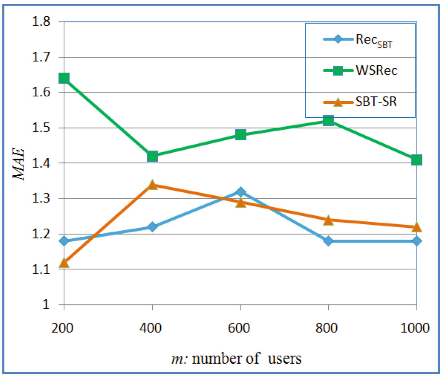

Fig. 4. Accuracy of three approaches

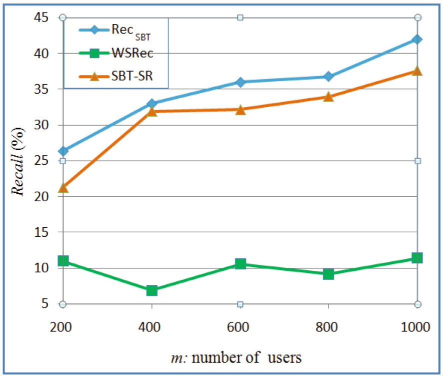

Fig. 5. Recall of three approaches

5 Related Works and Comparison Analyses

Collaborative Filtering (i.e., CF) has been proven a feasible resolution for service recommendation. Many researchers have investigated this recommendation problem and put forward various CF recommendation approaches, e.g., user-based CF [8], item-based CF [9] and hybrid CF [10], as well as their variants [11, 12].

However, the above works often assume that target user has similar friends or target service owns similar services, without considering the exceptional scenarios where neither similar friend (of target user) nor similar service (of target service) exists. In view of this shortcoming, *WSRec* approach is put forward in [7], where average rating of the services invoked by target user is recruited for service recommendation of target user. However, the recommendation accuracy of *WSRec* is often not high because of the adopted average idea. To improve the recommendation accuracy, *SBT-SR* approach is brought forth in our previous work [4], where "enemy's enemy is a friend" rule is employed for service recommendation. To further improve the recommendation performance, a novel recommendation approach named Rec_{SBT} is introduced in this paper, where more hidden social relationship information is taken into consideration, e.g., "enemy's enemy is a friend" rule, "friend's enemy is an enemy" rule and "enemy's friend is an enemy" rule. Through these social rules, more "possible friends" of target user could be found and correspondingly, more services that may be preferred by target user are recommended to the target user. At last, we validate the feasibility of our Rec_{SBT} approach through a set of simulation experiments deployed on well-known MovieLens-1M dataset.

6 Conclusions

Due to the inherent sparsity of user-service rating data, it is possible that the target user has no similar friends and the target service (i.e., the service preferred by target user) owns no similar services. In this situation, traditional CF recommendation approaches fail to deliver a satisfying recommendation result. In view of this shortcoming, we put forward a novel service recommendation approach Rec_{SBT} based on Social Balance Theory. Finally, a set of simulation experiments are deployed to validate the feasibility of our Rec_{SBT} approach in terms of recommendation accuracy and recall. In the future, we will introduce the time factor into service recommendation, so as to accommodate the dynamic waves of user preference.

Acknowledgements. This paper is partially supported by Natural Science Foundation of China (No. 61402258), Key Research and Development Project of Jiangsu Province (No. BE2015154), China Postdoctoral Science Foundation (No. 2015M571739), Open Project of State Key Laboratory for Novel Software Technology (No. KFKT2016B22), Open Project of State Key Laboratory of Software Engineering (No. SKLSE2014-10-03).

References

1. Wu, X., Cheng, B., Jun, L.C.: Collaborative filtering service recommendation based on a novel similarity computation method. IEEE Trans. Serv. Comput. doi:10.1109/TSC.2015.2479228
2. Xiang, Y.T., Jie, Z.: Dynamic personalized recommendation on sparse data. IEEE Trans. Knowl. Data Eng. **25**(12), 2895–2899 (2013)
3. Cartwright, D., Harary, F.: Structural balance: a generalization of Heider's theory. Psychol. Rev. **63**(5), 277 (1956)
4. Qi, L., Zhang, X., Wen, Y., Zhou, Y.: A social balance theory-based service recommendation approach. In: Yao, L., et al. (eds.) APSCC 2015. LNCS, vol. 9464, pp. 48–60. Springer, Heidelberg (2015). doi:10.1007/978-3-319-26979-5_4
5. Joseph, L.R., Alan, W.: Nicewander: thirteen ways to look at the correlation coefficient. Am. Stat. **42**(1), 59–66 (1988)
6. MovieLens-1M. http://www.grouplens.org/datasets/movielens/
7. Zi, B.Z., Hao, M., Michael, R.L., Irwin, K.: QoS-aware web service recommendation by collaborative filtering. IEEE Trans. Serv. Comput. **4**(2), 140–152 (2011)
8. Lin, S.Y., Lai, C.H., Wu, C.H., Lo, C.C.: A trustworthy QoS-based collaborative filtering approach for web service discovery. J. Syst. Softw. **93**, 217–228 (2014)
9. Li, D., Chen, C., Lv, Q., et al.: An algorithm for efficient privacy-preserving item-based collaborative filtering. Future Gener. Compt. Syst. **55**, 311–320 (2016)
10. Cao, J., Wu, Z., Wang, Y., et al.: Hybrid collaborative filtering algorithm for bidirectional web service recommendation. Knowl. Inf. Syst. **36**(3), 607–627 (2013)
11. Ming, D.T., Yu, X., Jian, X.L., Zi, B.Z., Frank, L.: Combining global and local trust for service recommendation. In: 21st IEEE International Conference on Web Services, pp. 305–312. IEEE Press, New York (2014)
12. Lian, Y.Q., Wan, C.D., Jin, J.C.: Weighted principal component analysis-based service selection method for multimedia services in cloud. Computing **98**, 195–214 (2016)