

Article

A Novel Approach for Web Service Recommendation Based on Advanced Trust Relationships

Lijun Duan ¹, Hao Tian ^{2,*}  and Kun Liu ²

¹ School of Computer, Hubei University of Education, No. 129, Gaoxin Second Road, Wuhan 430205, China

² School of Information and Communication Engineering, Hubei University of Economics, No. 8, Yangqiaohu Ave., Wuhan 430205, China

* Correspondence: haotian@whu.edu.cn; Tel.: +86-27-81973951

Received: 14 May 2019; Accepted: 24 June 2019; Published: 6 July 2019



Abstract: Service recommendation is one of the important means of service selection. Aiming at the problems of ignoring the influence of typical data sources such as service information and interaction logs on the similarity calculation of user preferences and insufficient consideration of dynamic trust relationship in traditional trust-based Web service recommendation methods, a novel approach for Web service recommendation based on advanced trust relationships is presented. After considering the influence of indirect trust paths, the improved calculation about indirect trust degree is proposed. By quantifying the popularity of service, the method of calculating user preference similarity is investigated. Furthermore, the dynamic adjustment mechanism of trust is designed by differentiating the effect of each service recommendation. Integrating these efforts, a service recommendation mechanism is introduced, in which a new service recommendation algorithm is described. Experimental results show that, compared with existing methods, the proposed approach not only has higher accuracy of service recommendation, but also can resist attacks from malicious users more effectively.

Keywords: Web service; recommendation; trust relationship; formalization; service popularity

1. Introduction

With the popularity of applications composed of services, the types and quantities of services have increased dramatically, and service-centered Internet is taking shape [1]. More and more web services that are emerging can meet the various needs of users, and the service information on the network is seriously overloaded. How to select services that meet the personalized needs of users from these massive services has become an urgent issue to be solved. Service recommendation technology is considered as one of the effective ways to solve the problem of service resource overload [2]. Web service recommendation refers to the process of pushing more suitable services to users according to a certain strategy among several candidate web services with the same functional attributes but different nonfunctional attributes. As an important means of providing personalized information services, service recommendation meets users' personalized needs by establishing a binary relationship between users and services, analyzing users' preferences and behavior characteristics, and recommending potential services that users are interested in.

Traditional collaborative filtering recommendation methods mainly calculate user preference similarity by comparing evaluation scores among users, and then recommend services for target users according to the index [3]. However, such methods often encounter data sparseness and cold start problems; moreover, they can do little to prevent attacks from malicious users who falsely evaluate the QoS (quality of service) value of web services, so their actual performance is not satisfactory. For this reason, many studies [4–8] have introduced trust relationships among users into recommendation

systems, believing that users tend to accept the recommended services from the friends they trust. Meanwhile, it is usually assumed that the preference similarity of trusted friends is also relevant [9]. However, these methods pay less attention to the impact of some typically available data such as service information and interaction logs on user preference similarity calculation, and they usually assume that user behavior is reliable and there is no malicious deception when judging similarity, so the accuracy and reliability of their recommendation results are often not satisfactory. Moreover, they usually fail to fully consider the dynamic characteristics of trust relationships, which results in the limitation of the accuracy of their recommendation.

To solve the above problems, in this paper, we present a novel approach which is different from existing methods in that: (1) Most of them only start with user-related data, but we take advantage of two typical data sources to optimize trust relationships: user interaction history and service popularity; (2) They seldom discuss the dynamics of trust relationship adequately, but we take the dynamic and pluralistic characteristics of the trust relationship into consideration; and (3) We put forward a recommendation mechanism with a more satisfactory ability to filter out malicious users. In this way, our proposed method makes more effective use of trust relationships to improve the performance of service recommendation. The experimental results show that the proposed method has better recommendation accuracy and stronger antifraud ability than the compared methods.

The rest of this paper is organized as follows. An overview of the existing research and the purpose of this paper are presented in Section 2. Section 3 details the formalization of the advanced trust relationships. A novel recommendation mechanism based on advanced trust relationships is described in Section 4 before conducting the experimental evaluations in Section 5. Finally, the conclusion is drawn in Section 6.

2. Related Work

Recent research shows that mining and leveraging user trust relationship to improve the performance of service recommendation is a feasible and effective method. In [10], a complementary service recommendation method which starts with the level of trust and the expert authority of trust is proposed. This method first realizes trust detection through the similarity of users' interests to narrow down the scope of recommended objects, and then solves the problem of professional disorientation through the recommendation of authoritative experts in specific professional fields. A novel trust recommendation model based on the latent factor model and trusty neighborhood fitting model is presented in [11]. It focuses on using rating features and attribute information to generate trusted service recommendations. By mapping attributes to features based on the filling method based on feature similarity and the filling method based on feature regression, the method can solve the failure problem of the latent factor model in the integrated model under cold-start situations. To filter the unreliable QoS data contributed by dishonest users, a trust-aware approach (TAP) for reliable personalized QoS prediction is proposed in [12]. Through calculating the reputation and similarity of users, the method can generate a set of trustworthy similar users and make a prediction for the active users by utilizing the QoS data and similarities. In [13], trust relationships were roughly divided into direct trust and indirect trust. Based on Bata Trust Model, the calculation method of trust degree was given. On this basis, a trust expansion model was presented to address the problem of sparseness. In combination with the user similarity, a trusted neighbor set construction algorithm was designed to assist a ranking algorithm in generating an optimal ranking model. The main factors affecting trust, such as the usefulness of recommendations, the transparency of the recommendation process, and the quality of the website, were analyzed and discussed jointly in [14]. Based on these three factors, a trust model with a three-layer structure was constructed in the paper. In combination with the model, the relative importance of these factors to trust building was experimentally evaluated. According to the experimental results, it was pointed out that transparency plays the most important role in trust building and recommendation quality. In view of the inconsistency of group decision-making caused by different opinions of experts in recommendation system, a trust-induced recommendation mechanism

was proposed in [15]. In this paper, an interval-valued trust decision space was defined to model the uncertainty of experts, which included the concepts of interval-valued trust function, interval-valued trust score, and interval-valued knowledge degree. Combining these concepts, group consensus strategy and negotiation mechanisms were proposed, and the ranking relation of interval trust functions was constructed to select the most suitable trust functions.

Although the above methods use trust relationships to improve the performance of service recommendation in varying degrees, they generally have two shortcomings: (1) They only consider the interaction records between users when defining the trust relationship, but in fact, the popularity of recommended services also affects the calculation of the trust degree. (2) When describing the dynamics of the trust relationship, most of them treat every recommendation equally, and seldom consider or quantify the actual effect of every recommendation on the positive or negative change of the trust relationship. Therefore, in order to overcome these shortcomings, on the basis of our previous research [16,17], we are motivated to consider more influencing factors, redefine the trust relationship, and propose a web service recommendation method based on advanced trust relationships to further improve the performance of service recommendation.

3. Formalization of Trust Relationship

This paper holds that the trust relationship between users should be gradually established through the recommendation interaction history between users, which can be represented by the degree of trust. Assume that service s_i is recommended to user u_a by user u_b , and s_i is invoked and evaluated by u_a in turn. If user u_a 's evaluation score on s_i exceeds u_b 's score on the service, it can be considered that this recommendation has a positive effect on u_a 's trust relationship to u_b ; then, the trust degree from u_a to u_b should be increased. On the contrary, if the evaluation value of u_a on s_i is lower than that of u_b , it can be deemed that this recommendation has a negative effect on u_a 's trust relationship to u_b , and then the trust degree from u_a to u_b should be reduced. If user u_a 's score on s_i is equal to that of u_b , it can be decided that this recommendation has the expected effect on u_a 's trust relationship to u_b , and then the trust degree from u_a to u_b should remain unchanged. Therefore, it can be concluded that the degree of trust between users is dynamically changing with the recommendation behavior between them.

Based on the above analysis, this paper argues that the degree of trust mainly depends on the user's experience of using the recommended service, that is, the degree of trust is related to the effect of each recommendation. At the same time, it also depends on the similarity of preferences among users and the popularity of recommended services. The higher the similarity between users' preferences, the more accurate the recommendation is; the lower the popularity of recommended services, the greater the influence of the recommendation effect on the degree of trust.

Hence, in this section, we first propose the definition of trust relationship, then discuss the main factors affecting trust relationship, and finally give the calculation method of trust degree.

3.1. Definition of Trust Relationship

In this paper, we mark the user set as U , the service set as S , and the trust relationship as T . Then, a trust relationship can be defined as a triple, which can be described as Equation (1).

$$T = (U, P, D) \quad (1)$$

where P means the set of trust paths, and D denotes the trust degree. They are the main factors that constitute the trust relationship.

In addition, based on the concepts of edge and path in graph theory, we define the length of trust path P as the length of the path connecting two users, that is, the number of edges, and mark it as $L(P)$. When $\min(L(P)) = 1$, we call T a direct trust relationship, and when $\min(L(P)) > 1$, we call T an indirect trust relationship. Figure 1 shows a sample of indirect trust relation between user u_a and user u_c . It is

easy to see that there are three trust paths from u_a to u_c , namely, $P_1: u_a \rightarrow u_b \rightarrow u_c$, $P_2: u_a \rightarrow u_g \rightarrow u_h \rightarrow u_c$, and $P_3: u_a \rightarrow u_f \rightarrow u_e \rightarrow u_d \rightarrow u_c$, and their lengths are respectively $L(P_1) = 2$, $L(P_2) = 3$, and $L(P_3) = 4$.

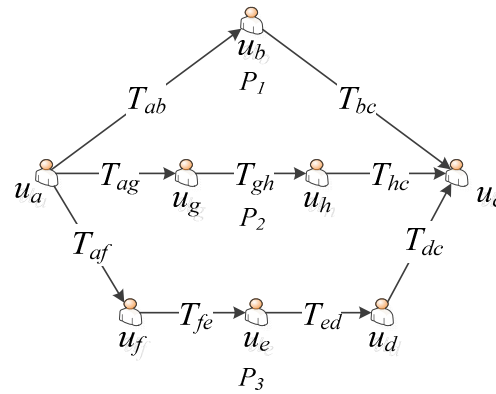


Figure 1. A sample of indirect trust relation.

Obviously, there is a difference in the calculation of D between the two types of trust relationships. In our previous study [17], indirect trust degree was calculated by selecting the maximum trust degree path without considering the influence of other trust paths, which may lose part of the trust information. Therefore, we put forward an improved calculation method of trust degree in this paper, which can be described as Equation (2).

$$D_{mn} = \begin{cases} \frac{\sum_{P_l \in P_{mn}} \frac{1}{L(P_l)} \left(\prod_{D_{ab} \in P_l} D_{ab} \right)}{\sum_{P_l \in P_{mn}} \frac{1}{L(P_l)}}, & \min(L(P_{mn})) \geq 1 \wedge L(P_l) \leq 5 \\ 0, & \text{else} \end{cases} \quad (2)$$

For two given users: u_m and u_n , D_{mn} is the trust degree of u_m to u_n . P_{mn} denotes trust paths from u_m to u_n , and P_l is one of them. D_{ab} is the trust degree on the edge from u_a to u_b in path P_l . In order to limit computing costs, we set the maximum length of a trust path to 5. It should be pointed out that $D_{ab} \neq D_{ba}$. As can be seen from Equation (2), this paper fully considers the impact of the main indirect trust paths in calculating the degree of indirect trust, and adopts the information entropy principle, which makes the measurement of indirect trust more accurate and credible. When there is only one trust path from u_m to u_n , D_{mn} is a direct trust degree, whose value can be calculated by Equation (6) below.

3.2. User Preference Similarity

In our view, users' similar preferences are the basis of their trust in each other, and the similarity of users' preferences is mainly measured by two indicators: the number of services that users have both invoked and their evaluation of these services. Jaccard similarity coefficient (JSC) and Pearson correlation coefficient (PCC) are the two most commonly used methods for calculating user similarity in recommendation systems, but they do not take into account the impact of overlapping scores among users on similarity, and can hardly deal with data sparsity. In our previous research [16], an improved approach for calculating user preference similarity was proposed and proved to be more effective than JSC and PCC, which can be shown as Equation (3).

$$Sim_{ab} = PCC_{ab} \times JSC_{ab} \frac{\sum_{s_i \in S_{ab}} \left(1 - \frac{|r_{ai} - r_{bi}|}{r_0} \right)}{|S_{ab}|} \quad (3)$$

In the above equation, Sim_{ab} is the user preference similarity between u_a and u_b . PCC_{ab} and JSC_{ab} are the preference similarity between u_a and u_b calculated based on the PCC and JSC methods,

respectively. S_{ab} is used to represent the set of services the two users have both invoked and $|S_{ab}|$ is the total number of Web services in S_{ab} . r_{ai} and r_{bi} denote the evaluation value on service s_i by user u_a and u_b respectively, and r_0 is the specified maximum evaluation value. Additionally, to formalize these values, r_{ai} and r_{bi} are normalized with integers in the range from 1 to 5; specifically, 5 is the best evaluation value and 1 means the poorest value. It is clear that Sim_{ab} depends on the difference between u_a 's and u_b 's evaluation of the same service. The smaller the difference, the higher the value of Sim_{ab} , and vice versa.

3.3. Service Popularity

As mentioned earlier, the factor of service popularity also needs to be taken into account when formalizing trust relationships. For a service with high popularity, if it is invoked multiple times and the evaluation values given by different users are relatively concentrated, then even if the evaluation values of u_a and u_b are consistent, it cannot indicate that the preferences of u_a and u_b must be similar. Therefore, it can be considered that the service has a small impact on the calculation of user preference similarity. Conversely, if u_a and u_b have the same evaluation for a service that is rarely invoked, they are more likely to have the same preference, that is, when calculating the preference similarity between u_a and u_b , the influence weight of such services should be greater.

Based on the above analysis, we quantify the popularity of service by the number of invocations and design the according metric and use them to update Equation (3) as Equation (4).

$$Sim_{ab} = PCC_{ab} \times JSC_{ab} \frac{\sum_{s_i \in S_{ab}} \left(\frac{1}{1 + \lg(1 + V_i)} \times \left(1 - \frac{|r_{ai} - r_{bi}|}{r_0} \right) \times \frac{\sqrt{(r_{ai} - \bar{r}_i)^2 + (r_{bi} - \bar{r}_i)^2 + 1}}{1 + |r_{ai} - r_{bi}|} \right)}{|S_{ab}|} \quad (4)$$

where, V_i is the total invocation times of s_i , and \bar{r}_i is the average evaluation value of s_i . Compared with Equation (3), Equation (4) adds two fractions to the numerator. The former one is used to quantify the impact of service invocation times on the calculation of user preference similarity. This paper argues that similarity is inversely proportional to the popularity of the web service. More specifically, the larger the number of invocations is, the smaller the value of this fraction will be, which means that in similarity calculation, the higher the popularity, the smaller the service contribution, and vice versa. The latter fraction is designed to measure the influence of the similarity of users' evaluation values on preference similarity calculation. It can be seen that, when u_a and u_b have the same evaluation of s_i and the evaluation value is consistent with the average evaluation score of the service, the value of this fraction is 1. However, when u_a and u_b give the same evaluation values, and the difference between that value and the average evaluation value is larger, the value of this fraction will be higher, which indicates that if users have the same evaluation of some service and the evaluation value is more different from the mainstream one, this service will be more important in the preference similarity calculation.

3.4. Dynamic Characteristics of Trust

With the occurrence of recommendation behavior among users, their trust relationships will also change. Specifically, the effect of each recommendation will affect the trust degrees between users. If u_b recommends a service to u_a , and u_a 's evaluation value is higher than u_b 's, then the trust degree of u_a to u_b will increase; if the value is consistent with that of u_b , the trust degree will remain unchanged; if the score is lower than that of u_b , then the trust degree of u_a to u_b will decrease. Hence, the dynamic characteristics of trust can be expressed as follows:

$$Y_{ab} = \begin{cases} 1 + \sum_{s_i \in E_{ba}} \frac{r_{ai} - r_{bi}}{r_0}, & E_{ba} \neq \emptyset \\ 1, & E_{ba} = \emptyset \end{cases} \quad (5)$$

In Equation (5), Y_{ab} denotes the dynamic characteristics of trust from u_a to u_b , and E_{ab} represents the set of services recommended by u_b to u_a . According to this equation, if the recommendation from u_b to u_a has a positive effect, the value of Y_{ab} will increase, which means that the trust degree of u_a to u_b will be enhanced. If the recommendation has the expected effect, then the value of Y_{ab} will be maintained, and the degree of trust will also remain unchanged. If the recommendation has a negative effect, the value of Y_{ab} will decrease and the degree of trust will be correspondingly weakened. In other words, the degree of trust between users will be dynamically adjusted with the effect of each recommendation.

Then, the direct trust degree from u_a to u_b can be calculated by Equation (6), and based on it, the indirect trust degree also can be determined by using Equation (2). It indicates that the degree of trust between two users is not only related to the similarity of their preferences, but is also affected by each recommendation behavior between them.

$$D_{ab} = Y_{ab} \times Sim_{ab} \quad (6)$$

4. Recommendation Mechanism

Once a user's trust relationship is established, the trustworthy users can be screened out, and the prediction of service evaluation value and recommendation behavior can be realized. In this process, the trustworthy user set of the target user is first established, and then the service evaluation value of the target user can be predicted based on this set.

In this paper, we adopt the strategy of setting threshold first and then limiting the scale to build a trustworthy user set. Specifically, for user u_a , a trust degree threshold D_T is first set; then, user u_n with trust degree D_{an} not less than D_T will be screened out, and the selected users will be sorted according to their trust degrees from the largest to the smallest. If the number of these users is too large, we set $|TU_a| = |TU|_m = n$ to select the first n users. Thus, u_a 's trustworthy user set TU_a is formed. Consequently, predicted evaluation values can be obtained by Equation (7).

$$P_{ai} = \bar{r}_a + \frac{\sum_{u_n \in TU_a} D_{an}(r_{ni} - \bar{r}_n)}{\sum_{u_n \in TU_a} D_{an}} \quad (7)$$

where P_{ai} represents the predicted evaluation value of target user u_a on service s_i , r_{ni} denotes the evaluation value of u_n on s_i , and \bar{r}_a and \bar{r}_n are the average evaluation values of all services invoked by u_a and u_n respectively. Equation (7) shows that the predicted evaluation value is mainly dependent on the average experience and trust relationships of the target user, as well as the specific evaluation value of each trusted user on the target service and his/her average experience.

Based on the above work, a web service recommendation algorithm based on advanced trust relationships (SRATR) is proposed, which can generate trustworthy user sets of target users by establishing trust relationships in the recommendation process to make trusted service recommendation for target users. The specific steps of SRATR can be described as the following Algorithm 1.

In this algorithm, the purpose of setting threshold P_0 and parameter k in the recommendation list is to ensure the accuracy of recommendation and control the number of recommended services. As can be seen from the above pseudocode, SRATR first builds the most trusted user group of the target user through his/her trust relationship, and then uses this user group to filter candidate services, thus giving reliable recommendation results. In addition, due to the existence of indirect trust relationship, SRATR can deal with the problem of sparse trust relationship caused by too few records of interaction between users in traditional methods. It can also be known from the above pseudocode that the time complexity of SRATR is $O(m \times n)$, where m denotes the number of users within the effective trust paths of the target user and n represents the number of services in the available services set.

Algorithm 1 SRATR**Input:** target user u_a , the set of users U , the set of Web services S and the evaluation value matrix R **Output:** recommended services list L_a

-
1. initialize the values of $|TU|_m$, D_T , P_0 and k
 2. $L_a = \phi$ and $TU_a = \phi$
 3. **for** each user $u_n \in U$ **do**
 4. calculate Sim_{an}
 5. calculate Y_{an}
 6. calculate D_{an}
 7. **if** $D_{an} \geq D_T$ **do**
 8. add u_n to TU_a
 9. **end if**
 10. **end for**
 11. **if** $|TU_a| > |TU|_m$ **do**
 12. sort u_m in TU_a according to D_{am} from the largest to the smallest
 13. keep the top $|TU|_m$ users in TU_a
 14. **end if**
 15. **for** each service $s_i \in S$ **do**
 16. calculate P_{ai}
 17. **if** $P_{ai} \geq P_0$ **do**
 18. add s_i to L_a
 19. **end if**
 20. **end for**
 21. sort s_i in L_a according to P_{ai} from the largest to the smallest
 22. keep the top k services in L_a
 23. **return** L_a
-

5. Experiments and Analysis

Experiments were conducted to evaluate the performance of the formalization of advanced trust relationships and the algorithm SRATR proposed in this paper, in which three similar methods are selected as comparison objects: IRATR [17], NCut [18], and SRMTC [19]. IRATR method is one of our previous research results and one of the foundations of this study. It can be used as a comparison object to test whether the performance of SRATR has been improved. NCut is essentially a user-based collaborative filtering method. As one of the traditional service recommendation methods, NCut uses the sorting method based on graph partition, which has better recommendation accuracy in the same category. Therefore, it was selected as the representative of traditional methods in this comparison. The SRMTC method also considers the influence of service attributes on trust relationships, establishes user trust relationships based on beta trust model, and builds trusted alliances of target users. Its principle is similar to SRATR, so its comparison can reflect the performance of our proposed method relative to similar approaches.

The experimental hardware environment in this paper consisted of a machine equipped with Intel Quad-Core i7 Processor at 3.40 GHz and 16 GB of RAM. The following experiments are carried out based on the Windows 10 operating system. In addition, Java7.0 is used for writing applications, Eclipse 4.5 for managing Web services, and MySQL 5.5 for managing user data.

5.1. Experimental Setup

In order to better reflect the comparison results, the same experimental environment as used in [19] was constructed. That is, Dataset 1 of WS-DREAM [20] was selected as the original test dataset, which comes from more than 20 countries and regions around the world and is stored in 150 distributed computers. This dataset is a widely recognized data source and has been selected as an experimental testing tool by many researchers. It contains 150 log files, and each log file records 10,000 invocations of

a user on 100 services. Each invocation record contains service QoS information such as response time, data size, and HTTP code. Table 1 shows details about some web services in Dataset 1, and Table 2 shows some samples of invocation results.

Table 1. Samples of details about some web services in Dataset 1.

Web Service ID	WSDL Address	Web Service Name	Provider Name	Location
8451	http://las.aei.polsl.pl/las2/las2.asmx?WSDL	Las2Service	polsl.pl	Poland
8460	http://streemo.pl/Portal/WebServices/SongService.asmx?WSDL	SongService	streemo.pl	Poland
8953	http://services.telekom.yu/WSResearch/registration.asmx?WSDL	Research	services.telekom.yu	Serbia and Montenegro

Table 2. Samples of invocation results.

Client IP	Web Service ID	Response Time (ms)	Data Size	HTTP Code
35.9.27.26	8451	2736	582	200
35.9.27.26	8460	804	14,419	200
35.9.27.26	8953	20,176	2624	−1

In this paper, response time is selected as the evaluation criterion of QoS, and the value of response time of each service is defined as the average of all response time in its invocation results. Thus, a 150×100 response time matrix (i.e., user-service matrix) formed by the results of 150 users invoking 100 web services can be obtained, as shown in Equation (8). Where, \overline{rt}_i^a represents the average response time when user u_a invokes service s_i .

$$RT = \begin{bmatrix} \overline{rt}_1^1 & \overline{rt}_2^1 & \cdots & \overline{rt}_{100}^1 \\ \overline{rt}_1^2 & \overline{rt}_2^2 & \cdots & \overline{rt}_{100}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \overline{rt}_1^{150} & \overline{rt}_2^{150} & \cdots & \overline{rt}_{100}^{150} \end{bmatrix} \quad (8)$$

In the experiment, 150 users are divided into two groups: training set and test set. Among them, 80% of users are included in the training set, and the remaining 20% are included in the test set. Users in the training set recommend services for users in the test set according to different recommendation strategies and methods in the following tests.

To better evaluate the quality of service recommendation, two measurement metrics are chosen: mean absolute error (MAE) and root mean square error (RMSE), which can be expressed as Equations (9) and (10), respectively. MAE is one of the widely used recommendation quality metrics in the recommendation system, which can intuitively reflect the actual situation of the prediction error. At present, most of studies adopt MAE as the criteria to measure recommendation quality and to evaluate the accuracy of service recommendation. RMSE is also usually used to measure the deviation between the prediction value and the evaluation score value in the evaluation of the quality of service recommendation. The combination of the two metrics can more clearly reflect the performance of service recommendation. By definition, the smaller the values of these two parameters, the better the quality of the recommendation.

$$MAE = \frac{\sum_{i=1}^N |P_{ai} - r_{ai}|}{N} \quad (9)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^N (P_{ai} - r_{ai})^2}{N}} \quad (10)$$

where, N represents the number of test samples, P_{ai} is the predicted evaluation value of target user u_a on service s_i , and r_{ai} denotes the actual evaluation value on s_i by u_a .

Additionally, the key parameters in SRATR are initialized as follows: $D_T = 0.3$, $P_0 = 0.35$ and $k = 5$, and the remaining comparison algorithms all use the optimal parameters described in the original literature. EXCEL is the instrument used for data and charts' analysis in this paper.

5.2. Comparison of Recommendation Quality

In this comparison, the number of trustworthy users is gradually increased from five to 30 at intervals of five. The differences in recommendation quality between the four methods are shown in Figures 2 and 3.

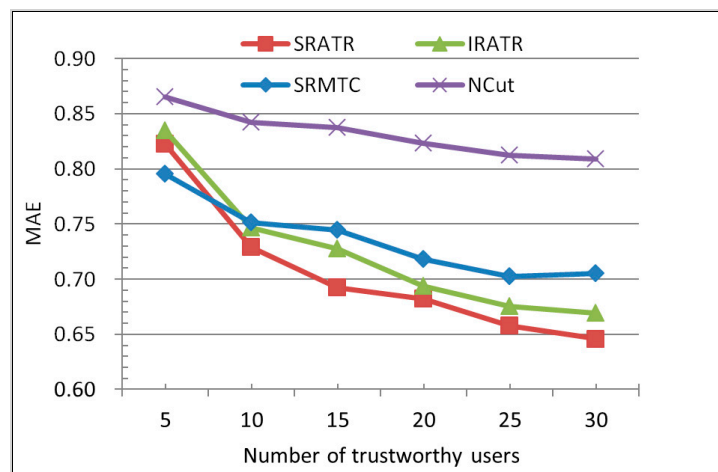


Figure 2. Mean absolute error (MAE) comparison without malicious users.

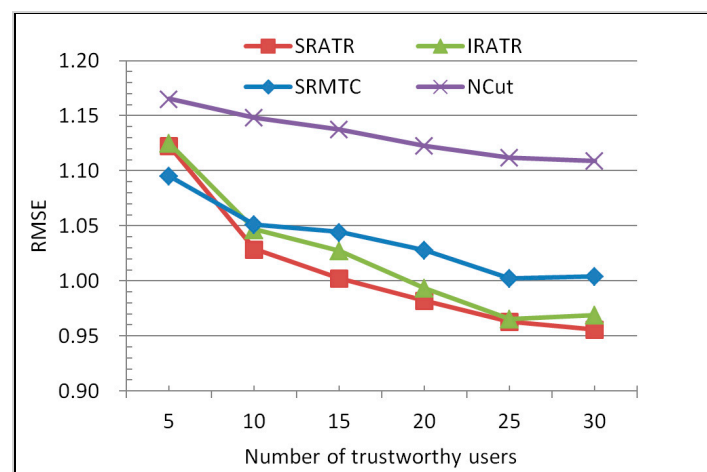


Figure 3. Root mean square error (RMSE) comparison without malicious users.

As can be seen in the figures, the MAE and RMSE values of the NCut method are always the highest when the number of trustworthy users is different, which is because the trust relationship between users is not considered in its algorithm. Although the ranking based on graph partition becomes more effective with the increase of trustworthy users and the MAE and RMSE values also show a steady decline, the overall performance is still not as good as that of the other three methods. As shown in the two figures, when the number of trustworthy users is not more than ten, the accuracy of recommendation of IRATR, SRMTC, and SRATC is not significantly different, which indicates that when the number of trustworthy users is relatively small, the trust relationship of the three methods is still unclear. It can also be seen from the figures that when the number exceeds ten and gradually

increases, the performance of the three is steadily improved. Among them, SRATR performs the best, followed by IRATR, and SRMTC is the worst. This proves that the performance of SRATR is significantly improved compared with IRATR after considering the influence of service popularity on trust relationship and optimizing the calculation method of indirect trust. Although the influence of service usage frequency and service reputation on trust relationship is also considered in the SRMTC method, its quantification method is too simple and will easily cause disturbance in the calculation of trust. Therefore, when the number of trustworthy users increases to 30, the performance of SRMTC becomes worse.

5.3. Comparison of Antifraud Ability

In order to verify the antifraud ability of these methods, the number of trustworthy users was set as 30 in this experiment and then the number of malicious users was gradually increased from zero to 30 at intervals of five. The changes of MAE and RMSE were observed and compared to check the antifraud ability of these methods. Specific experimental results are shown in Figures 4 and 5.

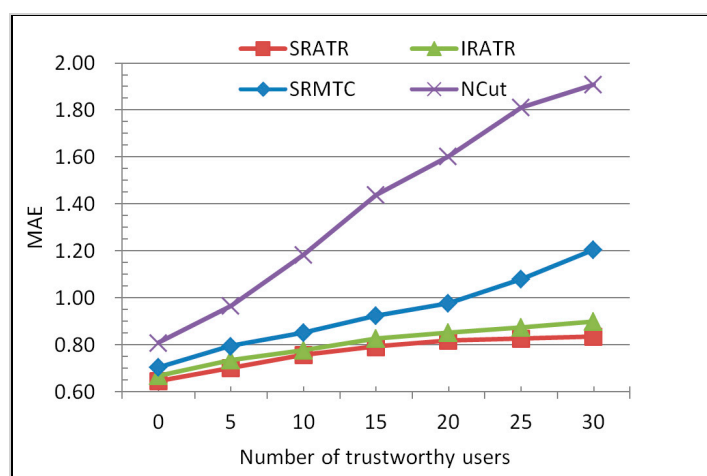


Figure 4. MAE comparison with malicious users.

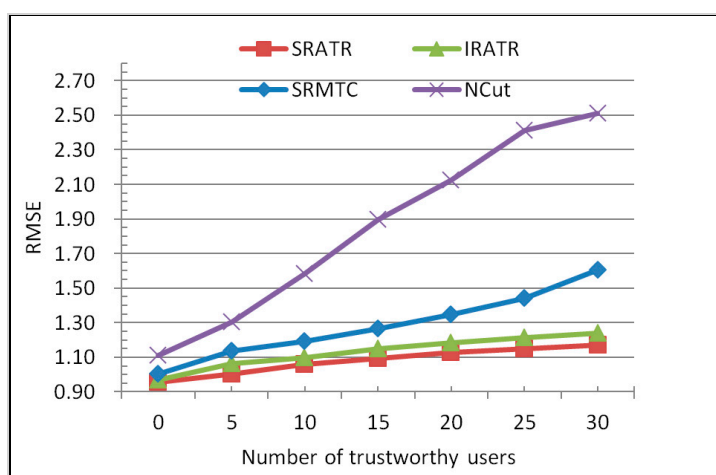


Figure 5. RMSE comparison with malicious users.

As can be seen from the figures, when the number of malicious users keeps increasing, the MAE and RMSE values of the four methods show an increasing trend to varying degrees, which indicates that the increase of fraud makes the quality of their recommendation constantly decrease. Among them, the MAE and RMSE values of the NCut method are always the highest, and the rise is also the fastest, because it can hardly distinguish fraudulent behavior in ranking, and the reverse adjustment in ranking

is insufficient. The remaining three methods all show good antifraud ability, which indicates that trust relationship has a good defensive ability against fraudulent behaviors in the process of service recommendation. However, the above two figures also show that the performance of IRATR and SRATR is always better than SRMTC's, regardless of the proportion of malicious users. Although the SRMTC method constructs a trustworthy alliance and can distinguish direct trust from indirect trust, it predicts the credibility of user behavior through probability density function and needs sufficient samples to judge malicious users, so it cannot react quickly to malicious behavior. On the contrary, IRATR and SRATR introduce dynamic adjustment mechanism in trust relationship modeling, and the emergence of each fraud will weaken the trust degree between users. Even if there are malicious users' attacks, the recommendation accuracy will not change much with the increasing proportion of malicious users. It can be further seen that SRATR always performs better than IRATR, which means that the advanced calculation method of indirect trust proposed in this paper, after considering the influence of multiple indirect trust paths, enhances the discrimination ability of trust relationship and can further resist attacks from malicious users.

5.4. Comparison of Efficiency

To verify the efficiency of the above four methods, their running times were compared. Since the user–service matrix is usually sparse in the practical application environment, in this experiment, some values are randomly removed from the matrix represented by Equation (8), thus forming five sparse matrices with densities of 0.04, 0.08, 0.12, 0.16, and 0.20, respectively. In addition, the previous experimental conditions remained unchanged, and the average running time of the different algorithms on the five matrices were recorded and calculated without considering malicious recommendations. The results are shown in Figure 6.

From Figure 6, it is evident that with the increase of matrix density, the running time of the four algorithms presents an upward trend. Among them, the running time of SRATR is longer than that of NCut, shorter than that of SRMTC, and roughly the same as that of IRATR. This is because NCut is essentially a traditional user-based collaborative filtering method which can directly utilize the QoS value in the dataset, so its running time is the shortest. SRMTC also uses trust relationships to generate recommendations, but in the process of its algorithm's execution, it needs to calculate the similarity between each candidate user and the target user in real time and determine the influence of each similarity on the prediction score. Therefore, SRMTC always has the longest running time. SRATR as an improvement on IRATR, and although new factors are taken into account and the calculation methods of some parameters are redesigned, it hardly adds any additional calculation cost.

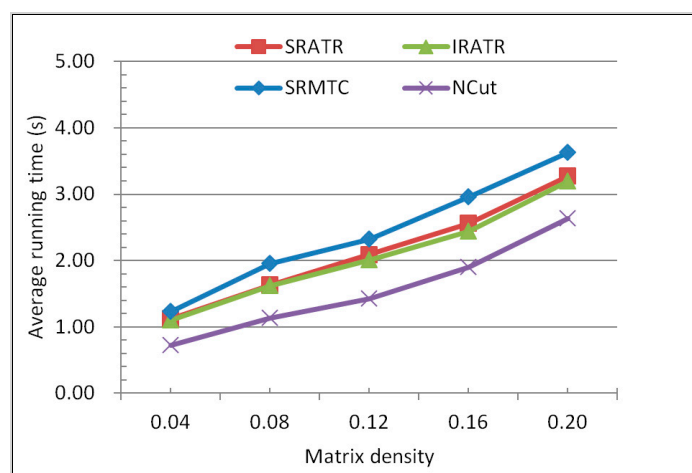


Figure 6. Comparison of running time.

6. Conclusions

In order to solve the problem of incomplete definition and insufficient dynamic characterization of trust relationships in existing methods of service recommendation based on trust relationships, in this paper, we presented a web service recommendation based on advanced trust relationships. By optimizing the calculation method of indirect trust, taking service popularity into consideration, and designing a dynamic trust adjustment method, the advanced trust relationship was formalized, and based on it, a service recommendation mechanism was designed and a novel service recommendation algorithm was proposed. Experimental results show that compared with our previous research, the proposed method not only ensures the efficiency of the algorithm, but also improves the quality of recommendation. Compared with other typical approaches, this method can provide more accurate recommendation results and has better anti-fraud ability. It should be noted that the use of typical factors affecting trust relationships is still insufficient in this paper, and the timeliness of QoS value of web services is not discussed in the research. In the following work, we will further optimize the trust relationship model by taking into account more factors, such as users' geographical location and interest transfer, to further enhance the recommendation accuracy, and we will consider the time effect of QoS value to further enhance the applicability of the model in dynamic environments.

Author Contributions: L.D. proposed the concept of research and collected data; H.T. performed the data analysis and wrote the manuscript; K.L. conducted the experiment.

Funding: This research was funded by National Natural Science Foundation of China under Grant No. 61572012.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, Z.; Wang, J.; Zhang, N.; Li, Z.; He, W.; He, K. A topic-oriented clustering approach for domain services. *J. Comput. Res. Dev.* **2014**, *51*, 408–419.
2. Pan, W.; Jiang, B.; Li, B.; Hu, B.; Song, B. Interactive service recommendation based on composition history. *J. Comput. Res. Dev.* **2018**, *55*, 613–628.
3. Meng, X.; Liu, S.; Zhang, Y.; Hu, X. Research on social recommender systems. *J. Softw.* **2015**, *26*, 1356–1372.
4. Wang, Y.; Yin, G.; Cai, Z.; Dong, Y.; Dong, H. A trust-based probabilistic recommendation model for social networks. *J. Netw. Comput. Appl.* **2015**, *55*, 59–67. [[CrossRef](#)]
5. Guo, G.; Zhang, J.; Yorke-Smith, N. A novel recommendation model regularized with user trust and item ratings. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 1607–1620. [[CrossRef](#)]
6. Azadjalal, M.M.; Moradi, P.; Abdollahpouri, A.; Jalili, M. A trust-aware recommendation method based on Pareto dominance and confidence concepts. *Knowl. Based Syst.* **2017**, *116*, 130–143. [[CrossRef](#)]
7. Park, C.; Kim, D.; Oh, J.; Yu, H. Improving top-K recommendation with truster and trustee relationship in user trust network. *Inf. Sci.* **2016**, *374*, 100–114. [[CrossRef](#)]
8. Aznoli, F.; Navimipour, N.J. Cloud services recommendation: Reviewing the recent advances and suggesting the future research directions. *J. Netw. Comput. Appl.* **2017**, *77*, 73–86. [[CrossRef](#)]
9. Zhu, W.; Zhong, Y.; Xu, J. Trustworthy service recommendation method based on social trust. *J. Chin. Comput. Syst.* **2017**, *38*, 503–508.
10. Kalai, A.; Zayani, C.A.; Amous, I.; Abdelghani, W.; Sèdes, F. Social collaborative service recommendation approach based on user's trust and domain-specific expertise. *Future Gener. Comput. Syst.* **2018**, *80*, 355–367. [[CrossRef](#)]
11. Li, W.; Mo, J.; Xin, M.; Jin, Q. An optimized trust model integrated with linear features for cyber-enabled recommendation services. *J. Parallel Distrib. Comput.* **2018**, *118*, 81–88. [[CrossRef](#)]
12. Su, K.; Xiao, B.; Liu, B.; Zhang, H.; Zhang, Z. TAP: A personalized trust-aware QoS prediction approach for web service recommendation. *Knowl. Based Syst.* **2017**, *115*, 55–65. [[CrossRef](#)]
13. Fang, C.; Zhang, H.; Zhang, M.; Wang, J. Trust expansion and listwise learning-to-rank based service recommendation method. *J. Commun.* **2018**, *39*, 147–158.

14. Nilashi, M.; Jannach, D.; Ibrahim, O.; Esfahani, M.D.; Ahmadi, H. Recommendation quality, transparency, and website quality for trust-building in recommendation agents. *Electron. Commer. Res. Appl.* **2016**, *19*, 70–84. [[CrossRef](#)]
15. Liu, Y.; Liang, C.; Chiclana, F.; Wu, J. A trust induced recommendation mechanism for reaching consensus in group decision making. *Knowl. Based Syst.* **2017**, *119*, 221–231. [[CrossRef](#)]
16. Tian, H.; Fan, H.; Du, W. Research on Web service discovery based on user community relations. *J. Commun.* **2015**, *36*, 28–36.
17. Tian, H.; Liang, P. Improved recommendations based on trust relationships in social networks. *Future Internet* **2017**, *9*, 9. [[CrossRef](#)]
18. Alejandro, B.; Parapar, J. Using graph partitioning techniques for neighbor selection in user-based collaborative filtering. In Proceedings of the Sixth ACM Conference on Recommender Systems, Dublin, Ireland, 9–13 September 2012; pp. 213–216.
19. Wang, H.; Yang, W.; Wang, S.; Li, S. A service recommendation method based on trustworthy community. *Chin. J. Comput.* **2014**, *37*, 301–311.
20. Zheng, Z.; Lyu, M. Collaborative reliability prediction for service-oriented systems. In Proceedings of the ACM/IEEE 32nd International Conference on Software Engineering (ICSE2010), Cape Town, South Africa, 2–8 May 2010; pp. 35–44.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).