A Trust Algorithm based on the Latent Trust from Users to Items

Meiyu Fang [1,2, a],Zhongkai Hu [2, b]

[1]School of Science and Technology, Zhejiang International Study University, Hangzhou, Zhejiang Province, P. R. China;

[2] College of Computer Science and Technology, Zhejiang University, Hangzhou, Zhejiang Province, P. R. China.

[a]hwdfmy@aliyun.com, [b]535300412@qq.com

Keywords: trust algorithm, referral trust, latent trust, customer-to-customer trust, customer-to-commodity trust, trust network.

Abstract. Trust relationship plays an important role in helping customers arrive at a trade decision in e-commerce. Numerous models and prediction algorithms have been proposed to calculate trust value. However, these models and algorithms are impractical in e-commerce transaction. In this paper, customer-to-customer and customer-to-commodity trust relationship sub-networks are proposed to get clear trust network which can be maintained easily. Trust relationships are divided into three types which are functional trust, referral trust and latent trust. We propose a trust algorithm to compute the trust value of customer-to-commodity trust which can help customers to make final purchase decisions directly. Our algorithm is based on more comprehensive trust features such as the referral trust, latent trust, domain similarity of Web commodities that users are interested in, and the influence of different reputations of users within two sub-networks. Experiment results with data sets from Epinions.com illustrate more accurate trust prediction compared with those existed algorithms.

Introduction

The rapid development of online social networks (OSNs) has enabled millions of people to interact daily with strangers. A complex trust network is established when users express their attitudes of trust or distrust online. Trust, as a means of social interaction, has attracted significant research interest in recent years [1–4]. Studies on trust have pervaded various application fields, from Web services[5] to e-commerce [6], recommender algorithms[7], and mobile social networks [8]. These methods combine the similarity and trust transitivity of usersto predict possible trust relationships previously unobserved in the network. To someextent, these methods have solved trust data sparseness in trust network studies. Theyhave also been used in applications to recommend new products to users and to lowerrisks involving the interaction between anonymous users. There are three classical studiesin this field. Guha et al.[10]proposed the popular TP algorithm, which serves asa suitable explanation for the emergence of new trust relationships in social networks;its “propagation” concept plays an important role in the evolution of the trust network.But it emphasis on only one trust feature, such as transitivity, ignoring the importantcharacter of fuzziness, and seldom considering the influence exerted by the differencein user reputation. The second representative studies are Common Neighbor(CN) algorithmwhich is proposed by Liben-Nowell and Kleinderg[9]and GTG(GenerateTrust Graphs) which is proposed by Jiang[17]. CN hypothesized that the likelihood ofan edge from user u_i to user u_k is proportional to the number of common neighborsof user u_i and user u_k. GTG only considered the users’ same trustees and reputationdifference while ignoring the trust influence of the trust networks. They only calculatecustomer-to-customer trust (i.e., user-to-user trust or u-u trust)resulted in the samesimple counting algorithm. The third method is SP(Statistical inference problem) algorithmwhich is a probabilistic trust propagation model that builds on the conceptof trust propagation proposed by Zhang et al.[1]. It improved the TP algorithm andproposed a new algorithm with better performance. The proposed model exploits themodern framework of probabilistic graphical models to formulate trust prediction asa statistical inference problem. SP has a better performance than TP and CN, but itsdirect inference and calculation on the information of OSNs usually make the trustnetwork too complicated to be obtained and difficult to be maintained. Its statisticalprocess is subject to probabilistic randomization, and its performance results are poor.

Considering these problems, we comprehensively considered many trust characters,such as transitivity and fuzziness, to optimize the trust relationship forecast ine-commerce. The trust network for latent trust relationships was then established andsimplified. Besides the similarity of users, we evaluated the latent u-i trust value interms of the differences of the trust relationships of users or latent users. The key pointto achieve our goal is calculating the similarity, reputation, differences of users' reputation,and determining the u-i trust network in which invisible trust relationships between users and items can be finding out.

Related work

As previously mentioned, in addition to analyzetrust network, we still concernthe similarity, reputation, and differences of users. Thus, related activities were performed around trustand its relationship's categories, trust network and user's reputation and so on.

Notably, one side trusts another in this study. One side is called a trustor, andthe other side is called a trustee. A trustor is most often a person, whilea trustee is a person or commodity.We divide trust into two categories according to the type of trustee in this study, namely,u-u trust and u-i trust.

u-u trust. If the trustorand trustee are both persons, then this type of trust is called u-u trust. u-u trust can be bidirectional because u_i can trust u_j, and u_j can trust u_i.

u-i trust. If the trustor is a person and the trustee is a commodity, service, or item, then this type of trust is called u-i trust. u-i trust should be unidirectional. If $u_i$trusts in i_j, then the direction is from u_i to i_j.

Trust is clearly different from reputation. The Oxford English Dictionary states that reputation is the common or general estimate of a person or thing with respect to character or other qualities. The reputation of a trustee is an aggregate value that comes from the trust degreeof all recommenders [11]. Therefore, trust expresses the possibility of individual-to-individual or individual-to-local trust (sometimes called local trust or local reputation in other literature [1,3,11]),whereas reputation expresses the possibility of a kind of global result.

Two actors are required for trust to exist because trust is a relationship between atrustor and a trustee (i.e., trust relationship). Jiang[17] divided trust relationships into two categories: referral trust and functional trust. This study extends Jiang's definition [17] and adds a type of trust relationship called latent trust. Functional trust represents the true ability of a target from his direct neighbor[12]. Referral trust represents the ability to directly recommend a suitable target, whereas latent trust represents the ability to indirectly recommend a suitable target. The trust value is attenuated by the extension of the link of trust propagation, in which1 is generally the maximum value,indicating complete trust, whereas 0 is the minimum value, suggesting the lack of a trust relationship. Each trust relationship has a fuzzy value called trust degree, which is larger than 0 and less than or equal to 1. The interweaving of millions of trust relationships on OSNs and e-commerce produces a complex network called the trust network.

Trust network. Trust relationship is combined into a complex network. It plays an important role in finding new information about an anonymous person or product. However, the network is extremely complex to maintain because each user can have hundreds of neighbors, and each of the neighbors of the trust chain will be fully extended again and again. In 2014, Jiang proposed a novel trust framework to address the accuracy calculation problem of trust prediction[17]. The issue of simplifying a complex trust network was effectively addressed by generating small trusted graphs for large OSNs, which can be used to improve the efficiency and practicality of previous trust evaluation algorithms. This approach was utilized in our work to determine the bridges between our target user u_i and item i_j.

Trust Algorithms. Trust value is derived from operating the trust and distrust matrix in the popular TP algorithm [10]. Ziegler et al. proposed the Tidal-trust prediction algorithm [13], which is a deep-first-search algorithm. Thus, the trust value between two nodes can be obtained by aggregating all user reviews searched from the source node to target node. Leskovec et al. used a directed symbol graph to represent a trust or distrust network[14]. Trust relationship was expressed by edge (u_1,u_2,+), whereas distrust was expressed by (u_1,u_2, -).Trust value was then calculated using structure balance theory. He et al. [6] verified that homogeneity does exist in the trust relationship. Similar users tend to establish a trust relationship, and trusted users tend to show more similarity. Xiao et al. [15] described trust relationship, including the trust network among users and reviewed networks among users and commodities. He proposed a trust prediction based on user similarity and global reputation according to sociology theory.

All the above mentioned algorithms have some reasonable improvement in the trust prediction research area. Two contributions have become the basis of the following studies. These contributions serve as a suitable explanation for the emergence of new trust relationships in social networks. The propagation concept has also contributed to the evolution of the trust network. However, trust prediction still requires additional practical improvements instead of studies that have thus far only considered transitivity. User reputation and its difference, fuzzy features, and trust influence from all latent trust relationships are important factors that are naturally considered in our work. The complexity of trust prediction can be lowered, accuracy of trust value can be improved, and trust fuzzy result can be in better line with the natural habits and expressive styles of people.

Problem description

Users express their attitudes of trust and distrust online, which establishes the user-to-user trust network. In addition, OSNs such as e-commerce websites permit their users to review Web items (e.g., purchased commodities) to help others make correct decisions. Users fetch things that are clearly valuable and trustworthy and give up things with bad reviews. The relationships between the users and Web items (or commodities) result in the u-i (commodity) trust networks. In this study, these two complex trust networks are called u-u and u-i sub-networks respectively. Let U represent the user set, U={u_1,u_2,u_3,u_4,u_5}, and I represent the item set, I={i_1,i_2,i_3,i_4},where the member is the service that users in U has accepted or the commodity that a user has purchased. Given that the vertices of U are common parts of the u-u and u-i networks that cannot be simply separated, these vertices are roughly marked using two colors (i.e., dark color for u-u and light color for u-i). The definitions of the two sub-networks in the real world are presented below (Figure. 1).

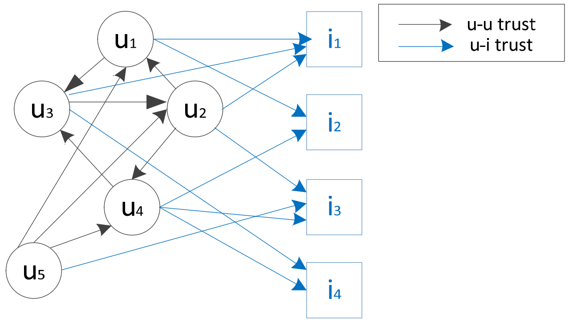


Fig. 1. u-u and u-i sub-networks.

u-u trust relationship and u-u trust network. The u-u trust relationship of u_i to $u_j$is denoted as (u_i, u_j,$T_{i,j}$), where u_i and u_j represent the user nodes of U. u_i, u_j∈U,$T_{i,j}$∈ (0,1). $T_{i,j}$represents the trust value in the degree that user u_i trusts user u_j. An edge from u_i to $u_j$exists, which is calledthe functional trust of u_i to u_j.

The u-u trust network is referred to as the u-u sub-network. It is denoted as $N_{u\text{-}u}=(V_{u\text{-}u}, E_{u\text{-}u})$, where $V_{u\text{-}u}$ is denoted as the set of user-to-user node pairs. Each node is a member of U. $E_{u\text{-}u}$ is an edge set whose elements are u-u trust relationships and$E_{u\text{-}u}=U\times U$.

u-i trust relationship and u-i trust network. The u-i trust relationship of u_i to i_j is denoted as (u_i, i_j,$R_{i,j}$), where u_i represents the user node of U and i_i represents the item node of I. u_i∈U, i_j∈I,

$R_{i,j}\in (0,1)$. $R_{i,j}$ represents the trust value in the degree that user u_i trusts item i_j. An edge from u_i to $i_j$exists, which is calledthe functional trust of u_i to i_j.

The u-i trust network is referred to as the u-i sub-network. It is denoted as $N_{u-I} = (V_{u-i}, E_{u-i})$, where $V_{u-i}= U \cup I$represents the set whose elements are user-item node pairs. E_{u-i} is an edge set whose elements are u-i trust relationships satisfiedwith the condition$E_{u-i}=U\times I$.

All the edges inE_{u-u} or E_{u-i}in Figure1are functional trusts. However, we are concernedwith the referral trust and latent trust in real life which are invisible in the graph of the said trust network. Edges are clearly hidden in the u-uor u-isub-network. Most previous studies have focused on the former. If users want to access a service or buy web products, then the u-i trust value can help them to make final decisions. For example, if the items are web commodities in e-commerce, $u_1$trusts ini_1,i_2,which results in the purchase of i_1 and i_2. Will u_1 buy i_3,i_4? Similarly, $u_5$trustsi_3, which results in the purchase of i_3. Will u_5 decideto buy i_1, i_2, i_4? Our problem is described as follows:

In the u-i sub-network, the functional trustorsof i_j(each trustor is referred to as fu_n and $(fu_i, i_j, R_{i,j})$ is a u-i trust relationship) constitute a set calledFU_j($FU_j \subseteq U$). Its definition is denoted in the following set:

$\mathrm{FU}_j= \{fu_1,fu_2,\ldots,fu_n\}$ (n = length of U, $i\leq n$)

Our predicted u-i trust value is the trust value from the users (u_is) to i_j,and theusers (u_is) are satisfied with the condition$u\in U \wedge u\notin FU_j$ to be more precise to the study problem. For instance, when j=4, i_j represents i_4, $\mathrm{FU}_4=\{u_3,u_4\}$, and $fu_1= u_3$, $fu_2 = u_4$, n=2 (Figure1). For $u_5 \notin FU_j$, trust for u_5 to i_4 represents our research goal to be predicted.

Our study objective is determining how to calculate the referral trust and latent u-i trust results (UITrust), such as u_5 to i_4. A novel framework called theUITrust Framework is thus proposed.

UITrust Framework

The UITrustFramework is built on the trust network. The similarity of users and reputation difference of recommenders are key points of the UITrust Framework to predict u-i trust value, referral u-i trust,and latent u-i trust. The main ideas are as follows: trust can be recommended and predicted; the more similarities users have, the more similar views on the same item are observed; and different reputations can lead to different or even opposite reviews of the same item. All trust and reputation values are too fuzzy to be utilized in the entire framework.

Among these points, the definition of referral u-i trust and latent u-i trust must be clear. Latent u-i trust that involves the entire social network is complex. Referral u-itrust and latent u-i trust are elaborated in this section for better understanding.

Two cases of the functional trustors of an item. Now, $FU_j=\{fu_1,fu_2,\ldots,fu_n\}$, $n\leq$length of U, whose members are users who functionally trust in item i_j. Users who are concerned with item $i_j$usually consider the view of users onFU_i. The relationship between FU_i's members and other members in Ubut not in FU_i becomes one of the crucial problems to be solved first. Users,u_is($u_i\in U\wedge u_i\notin FU_j$)are divided into two cases:

(1) The first case is $FUR_j(FUR_j\subseteq FU_j)$:$u_i$ functionally trusts a user fur_k,user fur_k functionally trust item i_j. All these fur_ks constitute a set FUR_j, and we deduced that referral trust was created from $u_i$to i_j through fur_k.$fur_k\in FUR_j$, $k\leq$ length of U. For example, see Figure 1.,u_5 has a referral trust to i_4 through the bridge $u4$,$u_4\in FUR_4$.

(2) The second case is$FUL_j(FUL_j\subseteq FU_j)$:user ful_ktrust in an item i_j.Althoughno direct edge from u_i touserful_k exists, pathscan be found in the u-u trust network from u_i to ful_k,, $k\leq n$. If the paths exist, a latent trust relationship will be derived from u_i to $i_j$throughful_ks, $ful_k\in FUL_j$. All these ful_ks constitue a set calledFUL_j.

Referral u-itrust. The referral trust of u_i to $i_j$is denoted as $(u_i, i_j,RT_{i,j})$, where u_i represents a user node of U, and i_j represents the item nodeof I. $u_i\in U$, $i_j\in I$, $RT_{i,j}\in (0,1)$. $RT_{i,j}$ represents the recommender trust value in the degreethat user u_i directly trustsu_k, and u_k directly trusts item i_j.This condition is called as the referral trust of u_i to i_j.

The first case of UITrust is the referral trust derived from functional trust propagation. Every social network user acts as a trustor who has his/her trustees as long as he/she interacts with others. Trust propagation states that if these trustees have a functional or referral trust relationship with one item, then the trustor has more or less a referral trust relationship with this item. This condition has a transitive property from user u_i to user fu_r, user fu_r to item i_j (which is called$RT_{i,j}$and explained in the following section in detail).

Latent u-itrust. The latent u-itrust of u_i to i_j is denoted as (u_i, i_j,$LT_{i,j}$), where u_i represents the user node of U, and i_j represents the item node of$I$. $u_i \in U$, $i_j \in I$, $LT_{i,j} \in (0,1)$. $LT_{i,j}$ represents the speculative trust value, a simplified u-u trust network called u-u trust graph exists from user $u_i$tou_k, and u_k directly trusts item i_j.We call this condition thelatenttrust of u_i to i_j.

The second case of UITrust$LT_{i,j}$ comes from the latent trust fromu_i to fu_l. Besides direct trust recommendation, we usually have a circle of acquaintances in the real world. Similarly, OSN users have their trust network for determining latent trust relationships. Figure2(a) shows that although u_5 doesnot have an edge to i_4, three paths from u_5 to $u_3$exist [Figures 2(b), 2(c), and 2(d)]. ThePSN[17] processing algorithmof Jiang shows that latent u-u trust can be derived [Figure2(e)]. Finally, the $LT_{5,4}$result can be predicted because of trust propagation. Its specific algorithm is detailed in the following section.

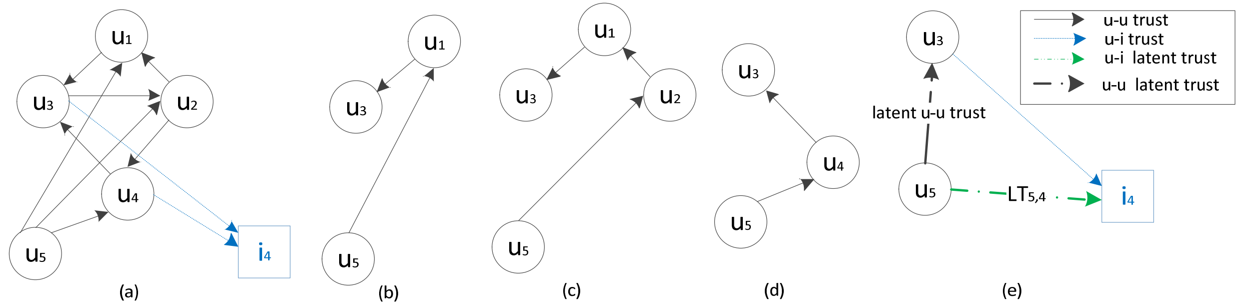


Fig. 2. Latent trust $LT_{5,4}$ formation mechanism.

Main idea of UITrust framework.u_i,$u_j \in U$. SupposeS_{u_i,u_j} represents the similarity between user i and user j. RD_{u_i,u_j}represents the influence led by the difference reputation ofu_i *and* u_j . The given conditions are as follows.

(1) Let LU be the length of U, LI the length of I, LFU the length of FU_j, $LFUR$ the length of $FURj$, and$LFUL$the length of FUL_j.

(2) $\exists u_i \in U, i_j \in I.\ \ i \leq LU,\ j \leq LI.$

(3) $\exists fur_k \in FUR_j$, $k \leq LFUR$, fur_k has a functional trust to i_j ,which is expressed as follows:

$$u_i \underbrace{\xrightarrow{\text{Referral Trust}}}_{fur_k} i_j.$$

(4) $\exists ful_k \in FUL_j$, $l \leq LFUL$, ful_k has a latent trust to i_j ,which is expressed as follows:

$$u_i \underbrace{\xrightarrow{\text{Latent Trust}}}_{ful_k} i_j.$$

The UITrustframework can be represented as the following function:

$$UIT_{i,j} = \begin{cases} \frac{\alpha S_{u_i,fur_k}+\beta RD_{u_i,fur_k}+\gamma RT_{i,k,j}}{m} & \text{if } \exists m paths,\ u_i \underbrace{\xrightarrow{\text{Referral Trust}}}_{fur_k} i_j. \\ \alpha S_{u_i,ful_k}+\beta RD_{u_i,ful_k}+\gamma LT_{i,j} & else\ \exists\ u_i \underbrace{\xrightarrow{\text{Latent Trust}}}_{ful_k} i_j. \end{cases} \tag{1}$$

$\alpha, \beta,$ and γarefactors for adjusting the importance degree of S,RD, RT, and LT. These factors are satisfied by the condition$\alpha+\beta+\gamma=1$. The resultscan be derived according to the following training experiments.

TALT:Trust Algoritms based on latent trust from users to items

User's similarity:S_{u_i,u_j}. User's similarityin $N_{u\text{-}u=}$ ($V_{u\text{-}u}$, $E_{u\text{-}u}$) is shown in the preference and specific domainof the user (e.g., age, interests, education, andreputation of users, and whether they have similar friends)[15]. In the e-commerce website in Epinions[16], other factorscan focus on how many similar friends and interests they have [17]. Thus,$S_{u_i,u_j}$is determined by the following equation:

$$S_{u_i,u_j} = \frac{|domain(N_{u_i}{'})\cap domain(N_{u_j}{'})|}{|domainN_{u_i}{'}\cup domainN_{u_j}{'}|} (2)$$

In Equation(2) where$N_{u_i}{'}$ denotes the trust node set of u_i'sneighbors in $N_{u\text{-}i}$, and$N_{u_j}{'}$ denotes the trust set ofu_j'sneighbors in $N_{u\text{-}i}$. N'_{u_i} and $N'_{u_j}$are strictly different fromN_{u_i} and N_{u_j}, whereu_i *and* $u_j$are trustors instead of trustees, and the neighbors in item set *I* does not include the distrust items. In $N_{u\text{-}i}$= ($V_{u\text{-}i}$, $E_{u\text{-}i}$), if the item's $rate \geq rate_{th}$($rate_{th}$ is a rate threshold that can be adjusted), thena trust relationship from user to item exists. Domain(N'_{u_i}) represents the domain (or itemcategory) set of every element in N'_{u_i}. |Domain(N'_{u_i}) ∩ Domain(N'_{u_j})|denotes the item number in the intersection. Eq. (2) shows thatthe situation can be drawn as follows:

It is clear that $0 \leq \frac{|domain(N_{u_i}{'})\cap domain(N_{u_j}{'})|}{|domainN_{u_i}{'}\cup domainN_{u_j}{'}|} \leq 1$, so,$0 \leq S_{u_i,u_j} \leq 1$.

Trust influence brought by the difference in the reputation of users: RD_{u_i,u_j}Reputation of the trustee plays an important role for trust computing. As the saying goes, "one who stays near vermilion gets stained red, and one who stays near ink gets stained." LetREP_{u_j} denote the global reputation of u_j, and $tr^i_{u_j}$ denote the trust degree of the *i*thneighbor to u_j. REP_{u_j}aggregates all $tr^i_{u_j}$from all u_j'sneighbors who trust u_j. If u_j's*i*thneighbor node u_i trusts u_j, then the value of $tr^i_{u_j}$ equals 1. Otherwise, this value equals -1 if u_j'sneighboru_i distrusts u_j. *m* is the number of total neighbors of u_j. Notably, REP_{u_j}produces a fuzzyresult between -1 and 1.

$$REP_{u_j} = \frac{\sum tr^i_{u_j}}{m} \quad i,j = 1,2,\ldots\ldots,m(3)$$

Referral trust calculation:$\mathbf{RT_{i,k,j}}$.

$RT_{i,k,j}$represents the referral trust from user *i* to item *j* through user *k*. Let $tr_{i,k}$represent the trust value of user *i*who trusts user *k*. $ra_{k,j}$ is the rate at which user *k* reviews item *j*. For the consistency with trust value in the range of [0, 1], $ra_{k,j}$is processed to $tr_{k,j}$usingthe following formula.$\mathrm{tr}_{k,j} = \frac{ra_{\mathrm{k,j}}}{\max_{k\leq LU, j\leq LI} ra_{\mathrm{k,j}}}$.

$$\mathrm{RT}_{i,k,j}= \mathrm{tr}_{i,k} \times \mathrm{tr}_{k,j} = \mathrm{tr}_{i,k} \times \frac{ra_{\mathrm{k,j}}}{\max_{k\leq LU, j\leq LI} ra_{\mathrm{k,j}}} (5)$$

Computation of latent trust $\mathbf{LT_{i,j}}$. The reference in Section 4.3 states that user u_k trustsitem i_j. $LT_{i,j}$ represents the latent trust from user *i* to item *j* through the u-u trust network of $u_k$when trust paths exist from u_i to u_k. The first step is to search all the trust paths along the chain of u_i to u_k to determine the $LT_{i,j}$ value.The second step is to compute the latent u-u trust value from u_i to u_k. The final result of $LT_{i,j}$from$u_i$to i_j can then be referenced.

Generating trusted graphs of Jiang. We introducedJiang's intuition as the first step [17]. When the user number increases, the complexity of the trust network clearly becomes difficult to control and all paths are exhausted. The generating trusted graphs of Jiang are adopted in this study through processing a large social network into a small one (PSN) [17], building the trust network (BTN) [17], and generating the trust graph (GTG processes) [17].

Algorithm of latent trust computation.An improved algorithm called TALTis proposed on the basis of Jiang's work for predicting the latent trust value $LT_{i,j}$. Our objective is to obtain the trust value of user u_i to item i_j, which is invisible. As previously described in Section 4.1, we first obtained the functional trustor set of item *i*: $FUL_j=\{ful_1,ful_2,\ldots\ldots,ful_n\}$ $(j=1,2,\ldots,LI;\ \ \mathrm{n}=LFUL)$. For each node $ful_k(k\leq n, ful_k\in FUL_j)$, we determined a trust graph *TG* between u_i and ful_k. Let *P* represent the paths set, and $TG_{i,k}$ represent the trust value computed for all path trusts of *P* according to Eq. (7). Given thatthe *kth* path:ps_k,$ps_k\in P, i\in N$, *i* is less than the length of the set *P*, the latent trust from u_i to i_j is created through the bridge $TG_{i,k}$. Deriving the value of $TG_{i,k}$ is the second thing for computing $LT_{i,j}$. Let tr_{ps_k}denotethe *k*th trust value of path ps_k, which is an element in *P*;tr_{ps_k} is composed of several trust edges labelede_n on path ps_k,e_n denotes node n_i to node n_j, and the *n*th edge (trust value) of ps_k, so $\mathrm{e_n}>0$. $tr_{ps_k}=\prod p(n_i,n_j)$. $\prod p(n_i,n_j)$denotes the value by multiplying every edge's priority on the path ps_k.

$$\mathrm{TG_{i,k}}=\frac{\sum \mathrm{e}_n}{m}=\frac{\sum tr_{ps_k}}{m}=\frac{\sum_k \prod p(n_i,n_j)}{m}(7)$$

In formular (7), k, n=1,2,…,m; n is the nth edge of $\mathrm{ps_{k.}}$Similar to the example in Figure3, suppose i=5, j=4, and after PSN and BTN, the TG and priority between each node on the trust graph is shown as the circle in Figure3. $\Psi=\{p1,p2,p3,p4\}$.p1 starts from u_5 to u_1 and then to u_3; p2 starts from u_5 to u_2, and thento u_1 and end with u_3; p3 starts from u_5 to u_4,end with u_3;and p4 starts from u_5 to $u_2$to u_4 to u_3.

$$\mathrm{TG_{5,3}}=\frac{tr_{p_1}+tr_{p_2}+tr_{p_3}+tr_{p_4}}{4}=\frac{0.7\times0.9+0.8\times0.6\times0.9+0.6\times0.7+0.8\times0.8\times0.7}{4}=0.4825$$

Third, $\exists u_k, u_k$is the bridge between u_i and i_j. $LT_{i,k,j}$ can be obtained as follows:

$$\mathrm{LT_{i,k,j}}=\mathrm{REP}_{\mathrm{u}_k}\times\mathrm{TG_{i,k}}\times\mathrm{tr_{k,j}}$$

$$=\begin{cases} 0\ REP_{u_j}<0 \\ \mathrm{REP}_{\mathrm{u}_k}\times\mathrm{TG_{i,k}}\times\frac{ra_{\mathrm{k,j}}}{\max_{k\leq LU,j\leq LI} ra_{\mathrm{k,j}}}\ REP_{u_j}\geq 0 \end{cases}(8)$$

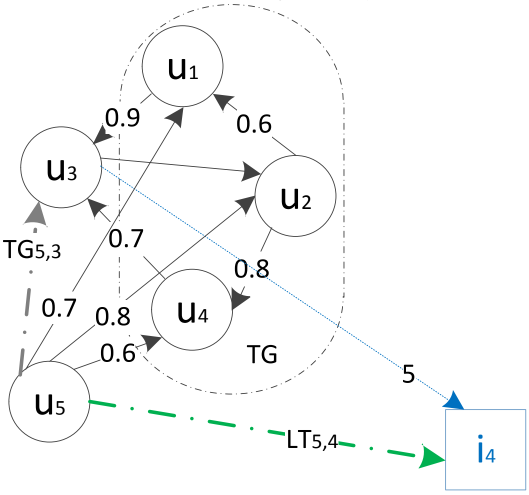


Fig. 3. Trust graph example.

UITrust algorithm: TALT. TheTALT algorithm was defined as follows (Table 1) according to Eq. (1). Let *G* be the same as algorithm 1.α, β and γare the parameters that can be adjusted in the following experiments.

Table 1.TALT algorithm**.**

```
TALT algorithm (prediction of synthesized trust from user i to item j: $UIT_{i,j}$)
Input (G, SOURCE, SINK) G = (V,E)
Input α 、β and $\gamma$ , α + β + $\gamma$ = 1, trust values, distrust values between users and users
  (1 denotes trust, whereas -1 denotes distrust), ratings (user ratings for item).
Output $UIT_{source,sink}$
$i \leftarrow$ the index of SOURCE in $U$; $j \leftarrow$ index of SINK in $I$
$FU_j = \{fu_1, fu_2, \ldots\ldots, fu_n\}$ (see Definition 5)
For each element $u_k$ of Set $FU_j$
    $REP_{u_k} = \frac{\sum tr_{u_k}^{i}}{m}$    $(i, j = 1,2, \ldots\ldots, \mathrm{m})$ (Equation 2)
    $S_{u_i,u_k} = \frac{|domain(N_{u_i}{}') \cap domain(N_{u_k}{}')|}{|domainN_{u_i}{}' \cup domainN_{u_k}{}'|}$(Equation 3)
    $RD_{u_i,u_k} = 10^{\delta * |REP_{u_i} - REP_{u_k}| * (|REP_{u_i} - REP_{u_k}| - 1)} \cos(\frac{|REP_{u_i} - REP_{u_k}|}{4}\pi)$(Equation 4)
end for
$FUR_j \leftarrow$ The Referral u-i Trust Subset of $FU_j$
$FUL_j \leftarrow$ The Latent u-i trust Subset of $FU_j$
$RT_{i,j} \leftarrow 0, k \leftarrow 1$
for Each Element of $u_k$ of Set $FUR_j$
    $tr_{k,j} = \frac{ra_{k,j}}{\max_{k \le LU, j \le LI} ra_{k,j}}$[see Equation (4)]
    $RT_{i,k,j} = tr_{i,k} \times tr_{k,j} = tr_{i,k} \times \frac{ra_{k,j}}{\max_{k \le LU, j \le LI} ra_{k,j}}$(Equation. 5)
    $RT_{i,j} = RT_{i,j} + RT_{i,k,j}$;  $k$++
end for
$RT_{i,j} = RT_{i,j}/k$
$LT_{i,j} = REP_{u_k} \times TG_{i,k} \times tr_{k,j}$(Equation8)
$UIT_{i,j} = \begin{cases} \frac{\alpha S_{u_i,fur_k} + \beta RD_{u_i,fur_k} + \gamma RT_{i,k,j}}{m} & \text{if } \exists\, u_i \underbrace{\xrightarrow{\text{Referral Trust}}}_{fur_k} i_j \;\; m\ is\ the\ length\ of\ fur_k. \\ \alpha S_{u_i,ful_k} + \beta RD_{u_i,ful_k} + \gamma LT_{i,j} & else\ \exists\, u_i \underbrace{\xrightarrow{\text{Latent Trust}}}_{ful_k} i_j. \end{cases}$
$UIT_{SOURCE,SINK} = UIT_{i,j}$
```

Experiments and Performance Evaluation

Experimental design. We used a classical evaluation technique in machine learning; namely, leave one out, to test the performance of our algorithms. If a user has rated an item (e.g., SOURCE and SINK), that rate is masked and trust is calculated through our TALT algorithm.The calculated value is then compared with the masked rate. We mainly consideredfour metrics: mean absolute error(MAE), precision, recall, and FScore(as described in the following Section).

Data set of experiments. Epinions is an online community website where users can write reviews about commodities and services, as well as rate other user reviews. The ratings for reviews are provided by customers who have read the reviews and have assessed the degree of usefulness of the reviews. We used the data set which is called Extended Epiniondataset[16]; it was experimented and verified in in literature[9, 17]for trust prediction studies. The data set contains 132,000 users who have issued 841,372 statements (717,667 trusts and 123,705 distrusts), 1,560,144 articles, and 13,668,319articleratings.

Data pre-process . Records that connect File 2 with File 3 in this study correspond to the u-i trust. The ratings that range in[1, 5] are the ratings used in the presentstudy. We Obtained the u-u trust network with the first file. Then the u-u and u-i trust networks can be immediately created. Using the file called mc.txt, domain formation and all the priorities among all trustors and trustees can be calculated according to Jiang's processing method. For convenience, the former 63,000 rows of File 3 and 67,000 rows of File 1 are selected for the initial data set. Approximately 931 pairs of u_i tou_k relationships can then be filtered out for research. All these $u_i$to u_k relationships can form millions of $u_i$to u_k to i_j (item j) relationships through joining computation. We select 99,000 rows of records for processing and finally derive 6,300 rows of records fortraining and verifying experiments by combining our algorithms and Jiang's generating trust graph processing algorithm. These data satisfy several conditions wherein user u_i directly trusts user u_k and item i_j, and referral trust or latent trust exists between u_i to u_k.

Evaluation Metrics. The MAE can be expressed as follows:

$$\text{MAE}=\frac{\sum_{i=1\ldots\text{n},j=1,\ldots,\text{n}}|UITP_{i,j}-tr_{i,j}|}{n}(9)$$

In Formula (9),n denotes the number of trust relationships,$UITP_{i,j}$ is the predicted trust value of useru_i to item i_j,and$tr_{i,j}$ is the actual value that corresponds to the rate at which user u_i reviews item i_j.

MAE01 (Mean Error): MAE01 is similarly defined as in other studies to compare our algorithm with TP,CN, and SP algorithms mentioned in Section 1. The trust value in other studies is quantified to 0 or 1, so $\text{Qua}(UITP_{i,j})$ is used to compute a fuzzy trust value to determine the trust value. Supposea threshold value th. if $UITP_{i,j} \geq th$, $\text{Qua}(UITP_{i,j}) = 1\ else\ \text{Qua}(UITP_{i,j}) = 0$.Thus, MAE01 can be derived by the following equation:

$$\text{MAE01} = \frac{\sum_{i=1\ldots\text{n},j=1,\ldots,\text{n}}|\text{Qua}(UITP_{i,j})-tr_{i,j}|}{n}\ (10)$$

Trust prediction precision (TPP):

$$\text{TPP}=\frac{TP}{TP+FN}\quad (11)$$

TPP is similarly defined as TPR which is trust prediction ratein previous studies [4, 17, 15]. True Positive (TP) denotes the number of trust relationships whose predicted and actual values are more than th. False Negative (FN) denotes the number of trust relationships whose predicted values are less than th and the actual value is more than th.

In this paper, Recall and FScore are defined as following:

$$\text{Recall}=\frac{TP}{PP}\quad (12)$$

$$\text{FScore}=\frac{2*Recall*TPP}{Recall+TPP}\quad (13)$$

TP is defined as above, and PP is the number of trust relationships whose predicted values are more than th.

Experimental parameter settings. TheTALTalgorithms are implementedusing Java language in the Windows platform. We adopt classical machine learning theory (training and verifying method). Among 99,000 records, 50% are used for training,while the rest are used for verifying.

Initially, let $x = |REP_{u_i}\text{ - }REP_{u_j}|$ in Equation (4),cos($\pi x/4$) is a decreasing function. However, x values usually focus around 0.2,which results inRD_{u_i,u_j}values near a fixed value and no discrimination. Thus, a decreasing function $10^{\delta*x*(x-1)}$is designed in the range [0, 1]. Let δ =4 to obtain $RD_{u_i,u_j}$with suitable discrimination.

Second, parameters$\alpha, \beta,$ and γdecides the importance of S_{u_i,u_j},$RD_{\boldsymbol{u_i,u_j}}$, and $RT_{i,j}$ or $LT_{i,j}$in Eq. (1). In the parameter setting experiments, when α , $\beta,$ and γ are fixed, MAE first appears to decrease and then increase. This scenario showsthat we can decide the range of α , $\beta,$ and $\gamma$to obtain better performance (i.e., $0.1 \leq \alpha \leq 0.5$, $0.1 \leq \beta \leq 0.4$, and $0.1 \leq \gamma \leq 0.65$). The value of $\alpha + \beta + \gamma$ must equal 1. We conducted experiments to compare the effects of the fixed parameters$\alpha, \beta,$ and γ on each metric to determine the values ofα, β, and γ. At this point, we setth=0.5 [17,15]to calculate TPP, Recall,FScore,and MAE01.

Table 2.The detail values of Recall, TPP, FScore, MAE, MAE01 when$\alpha = 0.36$ $\beta = 0.19$ $\gamma = 0.45$.

Number	Recall	TPP	FScore	MAE	MAE01
750	0.961538462	0.862068966	0.909090909	0.461952165	0.17
800	0.979166667	0.87037037	0.921568627	0.454096452	0.161818182
850	0.985714286	0.873417722	0.926174497	0.441865491	0.15
900	0.989473684	0.903846154	0.944723618	0.411830251	0.120952381
950	0.991631799	0.918604651	0.953722334	0.404262176	0.111538462
1000	0.992982456	0.918831169	0.954468803	0.40471413	0.110967742
1050	0.994029851	0.930167598	0.961038961	0.399384982	0.098888889
1100	0.994805195	0.93872549	0.965952081	0.397288567	0.089756098
1150	0.995402299	0.945414847	0.969764838	0.38751905	0.085217391
1200	0.99556541	0.883858268	0.936392075	0.39309811	0.142352941
1250	0.99592668	0.876344086	0.932316492	0.396553396	0.148928571
1300	0.996303142	0.886513158	0.938207137	0.396544892	0.140376432
1350	0.996615905	0.895136778	0.943154524	0.392604684	0.137359242
Average	0.9899	0.9003	0.9428	0.4109	0.1283

Metric evaluation and comparison of algorithms. Table 2 shows the detail values of Recall, TPP, FScore, MAE, MAE01. when$\alpha = 0.36$ $\beta = 0.19$ $\gamma = 0.45$.The precisions as Recall, TPP and Fscore have increased and the absolute mean error MAE and MAE01 have a decreasing change. The performance improved(precisions became higher and absolute errors became lower).All metrics became stable with the increasing of the number of trust relationships. Precision (TPP) is high with the minimum TPP value of 0.86, indicating that our algorithms have high quality in terms of predicting u-i trust. Recall is higher than 0.96 and became stable at 0.99 with the verified data increasing to 1350. These results show that our methods can largely decrease the sparsity of trust data. We also noticed that FScore had the same condition as Recall and TPP. The trust network had a large data amount without loss of generality. Therefore, TALT has high-quality TPP of 0.9003,FScore of 0.9428, and recall of 0.9899. MAE of TALT is basically maintained at approximately 0.4109, and MAE01 is 0.1283.

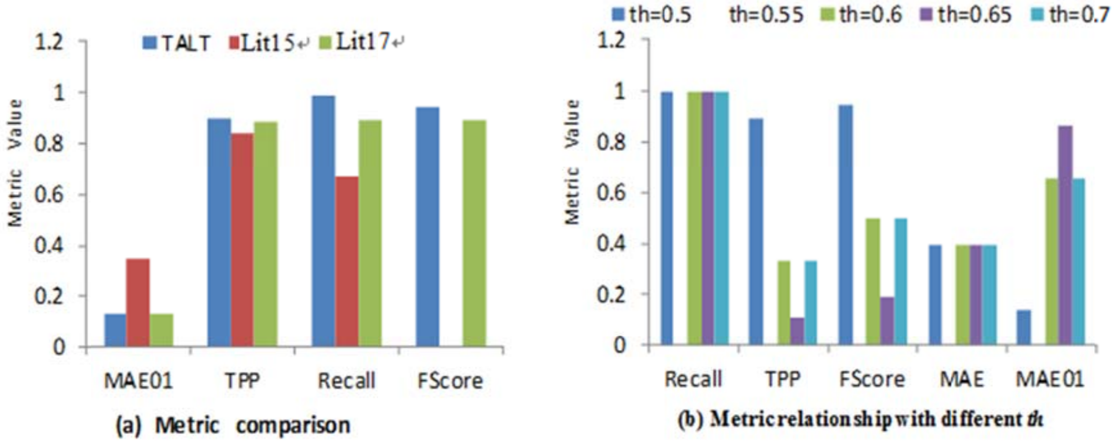
Performance comparison. Table 3 and Figure4(a) showa comparison ofprevious literatures[17] and[15].When TALT runs stably, MAE, TPP, Recall, and FScore are also stable. We used the stable metrics value. Notably, TPP is called TPR and Recall is called DPR in literature[15]. Previous literature lists 32 types of metric values with many different conditions and algorithms[17]. We only take the Minimum–Maximum algorithm of Jiang's heterogeneous settings (Table 6 in Jiang's paper). TALTclearly has better performance in the three metrics (i.e., TPP, Recall, and FScore) [17] and a previous algorithm [15]. Although the MAE of TALTis higher than0.4, our MAE is more trustworthy than the MAE in other studies to an extentbecause it is based on the fuzzy trust value.Its process doesnot include the conversion from vague to certainty quantity. We explain the reason for this condition with an example. If the predicted trust values are {0.2, 0.4, 0.6, 0.9} and the corresponding actual trust values are {0, 1, 1, 1}, then MAE= $(0.2 + 0.6 + 0.4 + 0.1)/4 = 0.325$according to our method.However, {0.2, 0.4, 0.6, 0.9} is first changed to {0, 0, 1, 1} using the other methods.The element value is converted to 1 when it is larger than a threshold that is usually 0.5. On the contrary, it is converted to 0 when it is less than the threshold. Thus, MAE=$(0 + 1 +$

$0 + 0)/4 = 0.25$. The same result {0.2, 0.4, 0.6, 0.9} possibly led to a different MAE. Our MAE01 is the lowest one compared with the Min-Max in previous literature [17, 15].

Table 3.Metrics Comparison.

Algorithm	MAE01	TPP	Recall	FScore
TALT	0.1283	0.9003	0.9899	0.9428
Literature 17 Min-Max	0.1346	0.8855	0.8946	0.8900
Literature 15	0.350	0.842	0.670	—

Experiments are performed to compare the proposed TALT algorithm with the three algorithms introduced earlier, namely, TPand SP. Figure4(b) shows that MAE is almost fixed when *th*varies. MAE01 largely and unstably changes. TALT runs best when th = 0.5 and decreases when *th* increases.



(a) Metric comparison

(b) Metricrelationship with different *th*

Fig.4. Metric comparison and relationship with different th

The TALTperformance is tabulated in Table 4 when we set th = 0.5.TALT achieves the best top-N precision when the parameters are set to $\alpha = 0.25, \beta = 0.3,$ and $\gamma = 0.45$. We search the parameter space for TP, CN, and SP for the settings that provide the best top-*N* precision (TPP in our study).TALT outperforms all other algorithms when *N* is larger than 200. SP is the second best algorithm. This condition demonstrates the effectiveness of the latent u-i trust prediction principle exploited in TALT.

Table 4. Top-N precision of TALT, th=0.5. (N is the number of Trust relationships)

alpha	beta	gama	N=100	N=150	N=200	N=300	N=500
0.36	0.49	0.15	0.3354	0.4952	0.5930	0.6558	0.7039
0.36	0.39	0.25	0.3228	0.4856	0.5853	0.6494	0.6983
0.36	0.29	0.35	0.3228	0.4856	0.5853	0.6494	0.6983
0.36	0.19	0.45	0.8734	0.9038	0.9186	0.9188	0.9302
0.36	0.09	0.55	0.8734	0.9038	0.9186	0.9188	0.9302
0.15	0.60	0.25	0.3354	0.4952	0.5930	0.6558	0.7039
0.25	0.50	0.25	0.3354	0.4952	0.5930	0.6558	0.7039
0.35	0.30	0.35	0.3228	0.4856	0.5853	0.6494	0.6983
0.55	0.20	0.25	0.1076	0.3221	0.3605	0.3701	0.4581
0.25	0.30	0.45	0.6835	0.7596	0.8062	0.8344	0.8575
0.25	0.40	0.35	0.5759	0.6779	0.7403	0.7792	0.8101
0.25	0.20	0.55	0.8734	0.9038	0.9186	0.9188	0.9302
0.15	0.50	0.35	0.3448	0.4352	0.5759	0.6779	0.7403
0.55	0.20	0.25	0.1076	0.3221	0.3605	0.3701	0.4581

Conclusion and future work

In this study,The TALT framework uses PSN to simplify a large social network into a small one that can beeasily maintained. This integrated framework reflects trust characters such as fuzziness, transitivity, and asymmetry. It fitsthe habitsof people inthe real world better than before. We design theTALT algorithm for this framework. The experiments with a data set from a real online commerce network validate the effectiveness of our work. Our algorithms have higher precision, Recall, and FScore than previous algorithms. Our main future work is toobtain a lower MAE and design new trust recommender methods thatcan overcome sparsedata.

Acknowledgements

This work is supported in part by the Ministry of Education, Humanities, and Social Sciences Research Project Fund of China (No.14YJAZH018), the National Natural Science Foundation of China (No. 61003254), Zhejiang Provincial Natural Science Fund of China (No.Y1080130), Zhejiang Provincial Natural Science Fund of China (No.Y1101304), and Zhejiang Provincial Department of Education Fund of China (No. Y201430471).

Reference:

[1]. RICHONG ZHANG, YONGYI MAO. Trust Prediction via Belief Propagation. ACM Transactions on Information Systems, Vol. 32, June 2014, pp. 15:10-15:27.

[2]. Jun Li, Xiaolin Zheng, Deren Chen. William Wei Song, Trust based Service Selection in Service Oriented Environment. International Journal of Web Services Research, July-September 2012; Vol.9, No.3, pp. 23-42.

[3]. ParthaSarathi Chakraborty, Sunil Karform. Designing Trust Propagation Algorithms based on Simple Multiplicative Strategy for Social Networks. Procedia Technology, Volume 6, 2012, pp. 534-539.

[4]. Yinhui He. Research on Trust Relationships in Online Social Networks [CHIESE].Electronically Technology University, March 2011.

[5]. Jun Li, Xiaolin Zheng, Deren Chen, William Wei Song, Trust based Service Selection in Service Oriented Environment, International Journal of Web Services Research, July-September 2012; Vol.9, No.3, pp. 23-42.

[6]. Meiyu Fang,Xiaolin Zheng,Deren Chen. A Reputation Evaluation Approach Based on Fuzzy Relation. International Journal of Computational Intelligence Systems, Vol. 4, No. 5, September 2011, pp. 759-767.

[7]. Surong Yan, Xiaolin Zheng, Yan Wang, Deren Chen, Exploiting two-faceted web of trust for quality enhanced recommendations.Expert Systems with Applications, 2013; 40:7080-7095.

[8]. Shuhong Chen, Guojun Wang, WeijiaJia. Cluster-group based trusted computing for mobile social networks using implicit social behavioral graph. Future Generation Computer Systems, Available online 12 June 2014; pp.110-128.

[9]. David Liben-Nowell, Jon Kleinberg. The link-prediction problem for social networks, J. Am. Soc.Inf. Sci. Technol. 2007, 58(7): 1019–1031.

[10]. Ramanathan V. Guha, Ravi Kumar. PrabhakarRaghavan, and Andrew Tomkins. Propagation of trust and distrust. In Proceedings of the 19th International Conference on World Wide Web,2004; pp.403–412.

[11]. J. Golbeck. Computing and applying trust in web-based social networks. Ph.D.Thesis, University of Maryland, 2005;pp.566-587.

[12]. Yao Y, Tong H, Yan X, et al. Multi-aspect+transitivity+bias: An integral trust inference model. Knowledge and Data Engineering, IEEE Transactions on, 2014, 26(7): 1706-1719.

[13]. Ziegler C N,Golbeck J. Investigating interactions of trust and interest similarity[J].Decision Support Systems,2007,43(2):460-475.

[14]. Leskovec J, Huttenlocher D, Kleinberg J. Predicting positive and negative links in online social networks[C].Proceedings of the 19th International Conference on World Wide Web.Raleigh, North Carolina,USA,2010:641-650.

[15]. Xiao Ma, ZaobinGan, Hongweilu, Yao Ma.Prediction of Latent Trust Relationships in E-commerce. Computer Science, Vol.41No.12,Dec 2014:138-142.

[16]. Wintersong. http://www.datatang.com/data/11850. Copyright ©2011-2014 datatang.com. Oct, 25, 2011.

[17]. Wenjun Jiang, Guojun Wang, JieWub Generating trusted graphs for trust evaluation in online social networks, Future Generation Computer Systems, 31 ,2014; 48–58.