


Article

A Hybrid Recommendation Method Integrating the Social Trust Network and Local Social Influence of Users

Lilei Lu ^{1,2,3} , Yuyu Yuan ^{1,2,*}, Xu Chen ^{1,2} and Zhaohui Li ⁴

¹ School of Computer Science (National Pilot Software Engineering School), Beijing University of Posts and Telecommunications, Beijing 100876, China; lilei_lu2013@163.com (L.L.); cx99@bupt.edu.cn (X.C.)

² Key Laboratory of Trustworthy Distributed Computing and Service, Ministry of education, Beijing 100876, China

³ Department of Computer Science, Tangshan Normal University, Tangshan 063000, China

⁴ Department of Resource Management, Tangshan Normal University, Tangshan 063000, China; li_zhao_hui@aliyun.com

* Correspondence: yuanyuyu@bupt.edu.cn

Received: 11 August 2020; Accepted: 7 September 2020; Published: 11 September 2020



Abstract: Recommendation system plays an indispensable role in helping users make decisions in different application scenarios. The issue about how to improve the accuracy of a recommendation system has gained widespread concern in both academic and industry fields. To solve this problem, many models have been proposed, but most of them usually focus on a single perspective. Different from the existing work, we propose a hybrid recommendation method based on the users' social trust network in this study. The proposed method has several advantages over conventional recommendation solutions. First, it offers a reliable two-step way of determining reference users by employing direct trust between users in the social trust network and setting a similarity threshold. Second, it improves the traditional collaborative filtering (CF) method based on a Pearson Correlation Coefficient (PCC) to reduce extreme values in prediction. Third, it introduces a personalized local social influence (LSI) factor into the improved CF method to further enhance the prediction accuracy. Seventy-one groups of random experiments based on the real dataset Epinions in social networks verify the proposed method. The experimental results demonstrate its feasibility, effectiveness, and accuracy in improving recommendation performance.

Keywords: recommendation method; social trust network; collaborative filtering; PCC; local social influence

1. Introduction

With the rapid development of network technology, various applications and corresponding online services become available to people in open and distributed environments such as e-commerce, social networks, cloud computing, Internet of Things, and so on. However, an enormous number of products and services make it difficult for users to search for what they desire most [1–3]. Under such circumstances, offering accurate recommendation information employing knowledge discovery techniques becomes very important and necessary in helping users make various decisions so that they can easily find items they want most from thousands of products and services [1,4]. In practice, recommendation systems have gained a rapid growth of attention as a tool to solve the problem of information overload for a wide range of applications in various fields [2,5,6]. Therefore, it is of significance for recommendation systems to provide users with appropriate and personalized recommendations about services, products, and information.

Many classifications of recommendation techniques including content-based, collaborative filtering (CF), knowledge-based, demo-graphic, and utility-based methods have been presented in recommendation systems [7,8]. CF is one of the most important and popular approaches in recommendation systems [1,5,9–11]. In essence, the CF recommendation method is based on the assumption that similar users usually hold similar preferences [1,3,6]. Recommendations are then made to the target user who needs to be offered prediction results according to the known information such as ratings acquired from reference users [3]. The reference users are usually referred to as a set of users who can offer useful reference information such as ratings, which can be employed to make prediction for unknown items of target users. Thus, the basic idea of CF-based recommendation systems is to search for the reference users of a target user at first, and then utilize their known information to make unknown prediction for the target user by means of similarity calculation including Pearson Correlation Coefficient (PCC), Cosine, Vector Space Similarity, and so on [12].

Although CF-based systems have the biggest advantage over content-based systems in that explicit content descriptions are not required, they suffer from several weaknesses concerning data sparsity, cold start problem on new users, and the difficulty in checking malicious or unreliable users [1,6]. These weaknesses would exert great influence on the performance of recommendation systems, especially the prediction accuracy. In essence, one of the possible reasons is that many recommendation systems ignore the social interactions and trust relationships among users [12,13]. The study shows that combining trust statements into the recommendation systems is one of the most effective and reliable ways to overcome the weaknesses of CF-based methods [1], however, to improve the prediction accuracy of unknown ratings is still one of the important challenging issues [4]. To explore a better solution to enhance the accuracy of rating predictions, a hybrid CF-based recommendation method is proposed in this study. The main contributions of our work are outlined as follows:

- The social trust network of users is combined into the proposed method in order to enhance the reliability of rating information source, which can also mitigate the cold start issue of recommendation systems to some extent.
- A two-step way of determining reference users is presented by employing direct trust between users in social trust networks and setting a PCC similarity threshold.
- An improvement in the traditional CF method based on PCC is presented to reduce extreme values of prediction, which is integrated into the proposed method to increase the accuracy and effectiveness of prediction.
- A local social influence (LSI) factor is introduced into the proposed method to further improve the prediction accuracy. The proposed LSI factor can also be regarded as a kind of trust weight in the social trust networks which do not explicitly define trust weight between users.

The rest of this paper is organized as follows. Section 2 describes the background and related work. Section 3 depicts the overview and Section 4 illustrates the main procedures of the proposed hybrid recommendation method. Section 5 introduces experimental objectives, preliminary work, demonstrates experimental methodology and a series of experiments and results, and finally presents respective analysis. Section 6 concludes the main work in this study and challenges in future work.

2. Background and Related Work

In this section, we first outline recommendation systems, then describe collaborative filtering recommendation systems and related work. Next, we investigate the concepts concerning trust, social trust network derived from social networks and their application in recommendation systems. Then, we introduce the concept of social influence in social network and explore its applications in trust-aware recommendation systems. Finally, we elicit our work in this study based on the above existing work.

2.1. The Outline of Recommendation Systems

In general, the recommendation systems are typically divided into three categories: CF-based, Content-based filtering, and the hybrid method [14]. The CF-based recommendation methods can be further subdivided into two basic categories: memory-based and model-based [1,15]. The memory-based methods are also called neighborhood or k-nearest neighbors methods [15], which usually make unknown rating prediction for a target user in terms of the rated items from reference users. In addition, they are either user-based or item-based [12], respectively based on user similarity and item similarity. The model-based methods typically employ machine-learning algorithms to train a predefined statistical model and predict unknown rating values for a target user based on the generated model [1,12].

In recent years, machine learning-based and context-aware recommendation methods have been surveyed and presented in existing work to enhance the recommendation performance [16,17]. Sarker et al. proposed a rule-based machine learning method to discover the behavioral rules of individual smartphone users to provide context-aware intelligent services [18]. Their method named “ABC-RuleMiner” can effectively identify redundancy in associations, and discover a set of non-redundant behavioral rules for individual users by taking into account the precedence of relevant contexts. McAuley et al. uncovered the implicit aspects hidden in users’ review text and combined these kinds of latent review topics with latent rating dimensions to develop a new recommendation model named “HFT”, by which they improved upon state-of-the-art models based on matrix factorization and LDA by 5–10% in terms of the mean squared error [19]. Moreover, an individual’s interests may not be static in the real world changing over time. Typically, recent behavioral patterns and corresponding machine learning rules are more meaningful than older ones when modeling and predicting users’ behavior. To deal with this issue, Sarker et al. proposed an approach named “RecencyMiner” for mining recency-based personalized behavior, utilizing individual’s contextual smartphone data, in order to build a context-aware personalized behavior prediction model [20].

Since the problem caused by cold start users is a serious barrier to the recommendation accuracy, there are some methods proposed to tackle this limitation by leveraging some side information. Cheng et al. proposed an aspect-aware rating prediction method to improve the accuracy of recommendations [21]. They developed a new topic model named “A³NCF” to extract typical features from reviews to guide its representation learning to capture a user’s special attention with an attention network. Afterwards, they leverage user reviews and item images to propose a multi-modal aspect-aware topic model to achieve good accuracy in rating prediction [22]. Yang et al. proposed HOP-Rec, a hybrid method involving random surfing that incorporates factorization and a graph model to harvest high-order information among neighborhood items for each user [23]. This method proves to be promising for general top-N implicit recommendation problems. Chen et al. proposed a unified CSE framework in which direct proximity and k-th order neighborhood proximity are differentiated in a user-item bipartite graph [24]. A proposed sampling technique is designed to acquire the two types of proximity relations, and extensive experiments showed its better performance based on eight benchmark datasets.

In this study, we concentrate on improving the user-based collaborative filtering method due to several reasons. One important point lies in the fact that we focus on the cold users who just started using a certain system and provided very few ratings. As mentioned in Ref. [25], these users tend to be a significant portion of the users. Secondly, the user-based CF method can be more conveniently applied to the online and real-time application scenarios because the related data can be stored in caches in the form of tables. Thirdly, the computation complexity of this kind of method may be lower compared with machine learning-based method which usually needs more iteration. Last but not least, this kind of method is often explainable, especially compared with matrix factorization, latent factor model, and so on.

2.2. Recommendation Systems Based on Collaborative Filtering

Collaborative filtering is one of the most popular and successful methods in recommendation systems [1]. The CF-based recommendation systems often utilize preference information such as ratings from a set of users i.e., the reference users (as mentioned in Section 1) similar to the target user to make unknown predictions [4,15]. Thus, to search for the reference users of a target user is often regarded as the most important step in the CF approach [4].

Since CF-based methods need the historical ratings of users [14,26], the most common metric employed in CF-based recommendation systems to find reference users is to resort to similarity measures [27,28]. PCC is one of the most popular similarity measures and empirical analyses have indicated that PCC outperforms other similarity measures in user-based CF [27,29,30]. It is often employed to directly calculate the similarity between the target user and each reference user [12]. Suppose that the total number of users in a recommendation system is n , then the PCC similarity between user i and user j can be described as the following Equation (1), where $Sim(u_i, u_j)$ denotes the similarity between user i and user j , $I(i)$, and $I(j)$ respectively represent the items rated by user i and user j , while k is the common item rated by user i and user j , $r_{i,k}$ and $r_{j,k}$ mean respective ratings of user i and user j to the common item k , \bar{r}_i and \bar{r}_j denote the average ratings of user i and user j . The resultant PCC similarities between each reference user and the target user are commonly considered as the weight of ratings from each reference user in the next prediction step. Then, the unknown rating for the target user can be acquired by means of combining the known ratings from all the selected reference users. This is the basic idea of the traditional CF method employing PCC:

$$Sim(u_i, u_j) = \frac{\sum_{k \in I(i) \cap I(j)} (r_{i,k} - \bar{r}_i) \cdot (r_{j,k} - \bar{r}_j)}{\sqrt{\sum_{k \in I(i) \cap I(j)} (r_{i,k} - \bar{r}_i)^2 \cdot \sum_{k \in I(i) \cap I(j)} (r_{j,k} - \bar{r}_j)^2}}, \quad (1)$$

$$(i, j = 1, 2, \dots, n)$$

The applications of CF in recommendation systems are quite popular in many fields, including online shopping, e-commercial services, social networking applications, and so on [11,14,31]. Many recommendation systems concentrate on finding the relevant items of each user only based on their interests, while neglecting the social interactions among users [12,13]. Boratto et al. [32] proposed a solution to remove sparsity from data by utilizing a user-based CF approach to predict the missing data points and improve the cluster performance in a group recommendation system. They presented k-means clustering to improve the accuracy of group recommendation and the results demonstrated the effectiveness; however, they ignored the interactions among users. Liu et al. [2] proposed a new user similarity model to improve the accuracy of CF method in the case of few ratings. They adopted both the local context information of user ratings and the global preference of user behavior in their model. However, they did not consider to combine the social relationship of users into the model. Ding et al. [33] proposed an improved PCC similarity method named “f-PCC” to predict the missing QoS (Quality of Service) values of cloud service for further evaluating cloud service trustworthiness. They introduced a correction parameter $C_{n,y}$ based on PCC to improve the prediction accuracy. Although the f-PCC prediction method is proved to be more accurate than other two existing methods, however, they did not take into account extreme values caused by the CF method based on PCC similarity calculation itself.

In practice, not every user can offer true and reliable ratings [1]. This situation will lead to inaccurate recommendation results. The fact that current studies often ignore the trust relationships among users may affect the quality of recommendations [34]. How to further improve the accuracy of a recommendation system has drawn too much concern of researchers and academia in various online application environments [4].

2.3. Trust and Its Applications in Recommendation Systems

Trust is a multidisciplinary concept concerning sociology, psychology, economics, computer science, and so on [35,36]. Although trust has been applied in many fields as a measurable concept, there is still no uniform definition [35,37]. Deutch et al. [38] presented the application scenario of trust when an entity falls into an uncertain position that it does not know how to handle, but did not define the trust itself. Adali et al. [39] believe that trust is a kind of relationship between a trustor and a trustee in the context of network, which shows the trustors' willingness in the vulnerable conditions of risk and interdependence. One of the most popular definitions derived from the sociologist Gambetta depicts that trust is a particular level of the subjective probability of one agent to assess another agent or a group of agents who will perform a particular action [40]. This reveals trust to be a one-way relationship between two parties in their interactions. Jøsang et al. [41] define trust as the extent to which one party is willing to depend on something or somebody in a given situation with a feeling of relative security, even though negative consequences are possible. From the definitions mentioned above, we can learn that trust has multiple features including subjectivity, uncertainty, context-dependence, one-way, and so on [35].

In both offline and online social networks, trust is an effective mechanism in all human interactions. It exerts an enormous impact on decisions whether to believe or disbelieve the information asserted by others [42,43]. In computer science, trust has been regarded as a measurable belief to help a trustor judge whether a trustee is trustworthy [44]. Trust has received more attention with the rapid development of the social networks in recent years. A social network is a social structure consisting of nodes which are generally persons or organizations and links acting as ties between these persons or organizations [45]. The network structure emanating from every individual accompanied with trust statement links has been regarded as a social trust network. Because people usually tend to trust their immediate neighbors over others when making decisions [42], it is easier to build trust through direct connections than indirect ones in social networks. The trust with direct connections in the trust network is usually referred to as direct trust. On the contrary, the trust with indirect connections in the trust network is often regarded as indirect trust or recommendation trust. Apparently, indirect trust can be developed through direct trust one after another along a certain path between any two users in social networks.

Since there is a close relationship between trust and social networks, researchers begin to concentrate on trust statements to increase the quality of recommendations [4]. Taking social trust between users in an online social network as additional inputs can improve the accuracy of the traditional recommendation system [46]. Trust-aware recommendation systems have been proposed to tackle the problems that exist in the CF approach employing information in social networks such as friendship and trust relationship among users [47]. The main idea is to use friendship or trust relations as additional information to offer high accuracy and personalized recommendations for the target user [5]. Chen et al. proposed a recommendation method using trust and distrust networks to address the cold start issue of new users [3]. However, whether the proposed method can extend to an e-commerce system deserves further research in future work. Zheng et al. presented a hybrid trust-based recommender system to mitigate the learning issues in online communities of practice and conducted a case study to testify the system [48]. However, the recommendation performance of the proposed system might be poor for novice learners without historical records. Moradi et al. proposed a reliability-based recommendation method to improve trust-aware recommender systems, which is proved to outperform several other methods in accuracy and cold start problem [1]. However, the trust between users is always kept the same and there is a static value for every user lacking explainable nature.

Although some trust-aware methods have been presented to enhance recommendation systems and made obvious achievement; however, the existing work is still not adequate. On the one hand, the trust weight between users is seldom assigned according to the social links. Obviously, the average weight assignment does not reflect the different trust degrees of a user to his different trusted neighbors.

On the other hand, the prediction accuracy for these systems can be further improved by taking into account reliability measures about the users and items [1,5]. In this study, we endeavor to find an effective hybrid way to solve the above two problems, in terms of the personalized direct trust between users which can be derived from social networks.

2.4. Social Influence and Its Applications in Recommendation Systems

In recent years, the rapid growth of social media including Facebook, Twitter, LinkedIn etc., introduces new opportunities for recommendation systems [47,49]. As described in the literature, social contextual information including friendship and trust between users has been demonstrated to be beneficial for recommendation systems [47,50,51]. For example, Seo et al. [47] introduced friendship strength to calculate the closeness between users in a social circle and proposed a friendship strength-based personalized recommender system in social networks. Ma et al. [50] presented employing a matrix factorization method to acquire the latent user vectors and item vectors. They combine social trust network of users into the proposed model as a regularization term to improve recommendation system. Ye et al. [51] proposed a collaborative recommendation framework that linearly merges ordinary user-based CF with only friendly-based CF.

However, most recommendation systems based on collaborative filtering assume that users are independent while ignoring the role of social influence in the process of users' decision-making [52]. In practice, the social influence plays an important role in our real life, which exerts a great effect on people's decision-making process. Ye et al. [49] believe that the social influence itself is a very general term related to different phenomena and quantitative, which can be regarded as the probability that a user adopts certain suggestions from his or her friends. Liu et al. [53] consider that the social influence is generally a measure, which represents the degree for one user to affect the behavior of others. They divide the social influence into two metrics: global and local. The quantification of the global social influence is usually quantified based on the entire social networks and hence it is very time-consuming. Conversely, the calculation of the local social influence is typically confined in a limited scope of social networks and thus is more time-saving, especially in searching reference users. According to Liu et al. [53], the formalization of the local social influence is defined as Equation (2), where $SI_{v_k,u}$ represents the social influence for node v_k to node u , $degree(v_i)$ denotes the node degree of v_i , and n is the total number of neighbors for node u :

$$SI_{v_k,u} = \frac{degree(v_k)}{\max\{degree(v_i), i = 1, \dots, n\}} \quad (2)$$

In fact, the social influence and its propagation has become an important topic in social networks. However, how to employ the users' social influence in recommendation systems to enhance its performance has not been given adequate attention in academia. Inspired by empirical roles of social influence in propagation via online social links, one important factor that we take into account in this study is the social influence among users. First of all, we take the above viewpoints from Ye and Liu et al. [49,53] and think that the social influence is a measure which can be quantitative by some reasonable method. Meanwhile, we distinguish the local social influence from the global social influence. Most important of all, we believe the social influence is of significance in recommendation systems and attempt to quantify the local social influence by employing the direct trust relationship between users. Finally, we employ it into our proposed hybrid method to make more accurate recommendation prediction.

Based on the above investigations and analyses from different perspectives, we propose a hybrid collaborative filtering recommendation method fusing social trust and local social influence of users in order to conquer the shortcomings caused by only focusing on a single perspective. The details of this method and key algorithms in each step will be depicted in the following sections.

3. Overview of the Proposed Hybrid Recommendation Method

In this section, we outline the general idea of the proposed hybrid collaborative filtering recommendation method. The overview of our proposed hybrid recommendation method is shown in Figure 1.

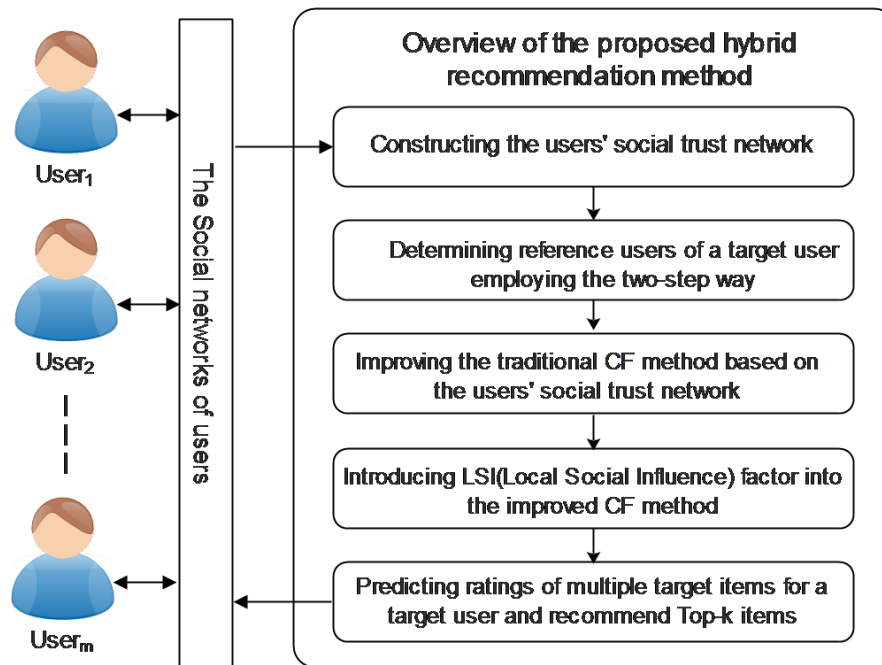


Figure 1. Overview of the proposed hybrid recommendation method.

To improve the performance of traditional CF-based recommendation systems, we attempt to construct a hybrid recommendation method from several aspects. Since the users' social trust network is a kind of important information source and an essential foundation of the recommendation method in this study, constructing the trust network based on direct trust relationships originated from the users' social networks becomes the first necessary task. Then, we propose a two-step way to determine reference users of a target user. Afterwards, we make an improvement for the CF method based on PCC and apply it in the users' social trust network. To further improve the prediction accuracy, we introduce the LSI (Local Social Influence) factor into the improved CF method. LSI can also be regarded as the trust weight of a trustor functioning in this study. Next, we bring about a hybrid recommendation method, which can be utilized to make unknown prediction for a target user about its unknown items. Based on the prediction results, Top-k items can also be acquired and recommended to the target user.

4. Main Procedures of the Proposed Hybrid Recommendation Method

In this section, we will illustrate the main procedures and describe the key algorithms in our proposed hybrid recommendation method according to the overview.

4.1. Constructing the Users' Social Trust Network

As is known to all, social networks can provide people with diverse and sufficient information about different products, services, and so on. In order to fully utilize the trust information derived from the users' social networks, we begin our method from constructing the users' social trust network based on their social networks. Chen and Massa et al. [3,25] have demonstrated that recommendation performance relies on the quality of the web of trust. To ensure that enough reference users can be

acquired, we suppose that users are encouraged as possible as they can to connect with each other via a trust network.

From the perspective of data structure, the users' social trust network can be regarded as a graph. Thus, constructing the users' social trust network is in essence equivalent to constructing a kind of graph. Since trust is considered as a one-way relationship as mentioned in Section 2.3, we construct the users' social trust network to be a directed graph, which is expressed as a triple tuple $G = (U, T, W)$ in this study. In the graph G , the element U, T, W respectively represents a set of users, a matrix of trust relationships and a matrix of trust weights, where $U = \{u_i | i = 1, 2, \dots, n\}$, each node u_i in U represents a user, n is the number of users; $T \subseteq U \times U$ and $T = \{t_{ij} | t_{ij} \in \{0, 1\}; i, j = 1, 2, \dots, n\}$, $t_{ij} = 1$ denotes user u_i trusts user u_j while $t_{ij} = 0$ denotes that there is no trust relationship between user u_i and user u_j , each directed edge (u_i, u_j) between users in G denotes the trust direction from a trustor user u_i to a trustee user u_j ; $W \subseteq U \times U$ and $W = \{w_{ij} | w_{ij} \in [0, 1]; i, j = 1, 2, \dots, n\}$, each element w_{ij} in W represents the trust degree from a trustor user u_i to a trustee user u_j . The trust degree from a trustor user u_i to a trustee user u_j , i.e., trust weight w_{ij} , which is determined by the local social influence of u_j to u_i in this study and denoted as $LSI(u_i \leftarrow u_j)$. Thus, $w_{ij} = LSI(u_i \leftarrow u_j)$. The definition and calculation of LSI factor will be described in Section 4.4.

In computer science, there are typically two methods to build the data structure of a graph: adjacency-matrix and adjacency-list. In this study, we select the adjacency-matrix method to generate the directed graph we desire. Each element (t_{ij}, w_{ij}) in the adjacency-matrix consists of two values, which respectively represent the trust relationship and the trust weight from a trustor named u_i to a trustee named u_j , as shown in Figure 2. The constructing process of the directed graph G representing the users' social trust network can be described as Algorithm 1. The construction of the users' social trust network first incorporates all of the pairs of direct trust relationships between users to acquire the user set U and the trust matrix T , then calculate LSI factors to form the weight matrix W based on the existing trust relationships.

4.2. Determining Reference Users of a Target User in the Social Trust Network

How to select appropriate reference users is a key issue that will affect the recommendation accuracy to a large extent. Different from the traditional CF method, we present a two-step way to select the final reference users for a target user who will be predicted the rating of an unknown item. The detailed descriptions are as follows.

	u_1	u_j	u_n
u_1			
u_i		(t_{ij}, w_{ij})	
u_n			

Figure 2. Adjacency-matrix representation of the directed graph of users' social trust network.

Algorithm 1: Constructing the users' social trust network

Input: the set of pairs of a trustor u_{tr} and a trustee $u_{te} : \{(u_{tr}, u_{te})\}$
Output: G

```

1  $U = \Phi$ ;
2  $T = 0_{n \times n}$ ;
3  $W = 0_{n \times n}$ ;
4  $inputSet = \{(u_{tr}, u_{te})\}$ ;
5 for each element  $(u_{tr}, u_{te}) \in inputSet$  do
6    $U = U \cup \{u_{tr}, u_{te}\}$ ;
7   for each element  $u_i \in U$  do
8     for each element  $u_j \in U$  do
9       if  $(u_i == u_{tr})$  and  $(u_j == u_{te})$  then
10         $t_{ij} = 1$ ;
11        update  $T$ ;
12      end
13    end
14  end
15 end
16 for each element  $(u_{tr}, u_{te}) \in inputSet$  do
17   for each element  $u_i \in U$  do
18     for each element  $u_j \in U$  do
19       if  $(u_i == u_{tr})$  and  $(u_j == u_{te})$  then
20        calculate the local social influence  $LSI(u_i \leftarrow u_j)$  of  $u_j$  to  $u_i$ ;
21         $w_{ij} = LSI(u_i \leftarrow u_j)$ ;
22        update  $W$ ;
23      end
24    end
25  end
26 end
27  $G = (U, T, W)$ ;
28 return  $G$ ;

```

4.2.1. Selecting Initial Reference Users of a Target User in the Social Trust Network

In this step, we determine the initial reference users of a target user based on their social trust network. We tend to utilize the direct trust relationship between users to achieve the goal. For one thing, as mentioned previously, it is indeed the real case that a user is more likely to believe statements from a trusted acquaintance than from a stranger [54]. For another thing, it has been shown that it is an effective way to utilize direct trust relationships to ensure the reliability of data source and enhance the accuracy of recommendation. If one user in the social trust network is trusted by the target user directly and meanwhile satisfies the next two conditions: (1) who has rated the target item which is unknown for the target user, (2) who has rated at least 2 common items with the target user, then this user is grouped into the initial reference users. Otherwise, this user does not belong to the set of initial reference users.

The detailed descriptions of this step are shown in Algorithm 2. In this algorithm, we introduce a new data structure, which is the rating matrix $M = (U_{id}, I_{id}, R)$. M represents the rating matrix of the historical rated items of all users, which has the same structure as the rating matrix in the Epinions dataset. Each line in M consists of three components: a user ID, a rated item ID, and its corresponding rating, which represents a piece of rating records of a user, corresponding to the element (u, i, r) in

M , where $u \in U_{id}, i \in I_{id}, r \in R$. U_{id} , I_{id} and R are three lists which separately consist of the user IDs, the rated item IDs, and ratings. Since one user may have rated more than one different items, the same user ID can appear more than once in U_{id} . Similarly, many users may have rated the same item, thus the same item ID can appear many times in I_{id} . In addition, different users may give the same ratings for different item IDs, the same ratings may appear many times in R . U_{dt} is used to store the direct trust users of the target user. N denotes the number of direct trust users in U_{dt} . $I_{u_{tr}}$ denotes the set of rated items of the target user. For each direct trust user of the target user, we initiate a list of sets to store the rated items separately in order to calculate the common items between each of them and the target user. U_{iRef} is to store the initial reference users who meet all the conditions mentioned in this step.

Algorithm 2: Determining the reference users of a target user—Step 1: selecting the initial reference users of the target user u_{tr}

Input: $G = (U, T, W)$, $M = (U_{id}, I_{id}, R)$, the target user u_{tr} , an unknown target item i_{un} of u_{tr}

Output: the set U_{iRef} of initial reference users of u_{tr}

```

1   $U_{dt} = \Phi$ ,  $U_{iRef} = \Phi$ ,  $n_{com} = 0$ ;
2  for each element  $u_i \in U$  do
3      for each element  $u_j \in U$  do
4          if ( $u_i == u_{tr}$ ) and ( $t_{ij} == 1$ ) then
5               $U_{dt} = U_{dt} \cup \{u_j\}$ ;
6          end
7      end
8  end
9   $N = |U_{dt}|$ ,  $I_{u_{tr}} = \Phi$ ,  $I_{u_{dt}}[N] = [\Phi, \Phi, \dots, \Phi]$ ;
10 for each element  $(u, i, r) \in M$  do
11     if  $u == u_{tr}$  then
12          $I_{u_{tr}} = I_{u_{tr}} \cup \{i\}$ ;
13     end
14     else
15         for each element  $u_j \in U_{dt}$  do
16             if ( $u_j == u$ ) then
17                  $I_{u_{dt}}[j] = I_{u_{dt}}[j] \cup \{i\}$ ;
18             end
19         end
20     end
21 end
22 for each user  $u_j \in u_{dtr}$  do
23      $n_{com} = |I_{u_{tr}} \cap I_{u_{dt}}[j]|$ ;
24     if  $i_{un} \in I_{u_{dt}}[j]$  and  $n_{com} \geq 2$  then
25          $U_{iRef} = U_{iRef} \cup \{u_j\}$ ;
26     end
27 end
28 return  $U_{iRef}$ ;

```

After performing Algorithm 2, we can obtain the set of the initial reference users of the target user. Next, we can continue Step 2 in order to obtain higher prediction accuracy.

4.2.2. Determining Final Reference Users by the Predetermined Threshold of PCC

To further improve the quality of the initial reference users, in this step, we first calculate PCC similarities between each of them and the target user, and then determine the final reference users based on the predetermined threshold of PCC similarities. In fact, the threshold of PCC similarities can either derive from the practical need of users, or be adjusted to an appropriate level employing machine-learning method according to the balance between cover rate of prediction and prediction accuracy in terms of MAE and RSME. Here, we won't go into it and regard it as a predetermined value. We will exemplify its positive effect on the prediction accuracy in a certain range of PCC, as shown in later experiments. The brief descriptions about how to select the final reference users of a target user are shown in Algorithm 3.

Algorithm 3: Determining the reference users of a target user—Step 2: determining the final reference users of the target user u_{tr}

Input: U_{iRef} , $M = (U_{id}, I_{id}, R)$, the target user u_{tr} , the threshold Sim_{thr} of PCC

Output: the set U_{ref} of final reference users of u_{tr}

```

1 calculate the average rating  $\bar{r}_{tr}$  of  $u_{tr}$ ;
2 for each element  $u_j \in U_{iRef}$  do
3   calculate the average rating  $\bar{r}_j$ ;
4   calculate the common rated items  $S_c$  between  $u_j$  and  $u_{tr}$ , and rating information  $S_{com}$  of  $u_j$ 
   using algorithm 4;
5 end
6 for each element  $u_j \in U_{iRef}$  do
7   calculate the similarity  $Sim(u_j, u_{tr})$  between  $u_j$  and  $u_{tr}$  using Equation (1);
8   if  $Sim(u_j, u_{tr}) < Sim_{thr}$  then
9      $U_{ref} = U_{iRef} - \{u_j\}$ ;
10  end
11 end
12 return  $U_{ref}$ ;

```

According to Equation (1) in Section 2.2, before calculating the PCC similarity between the target user and each of the initial reference users, the average ratings about their respective historical ratings should be computed. In fact, the average ratings can be achieved separately in terms of their own historical ratings by searching the rating matrix M . Meanwhile, the common rated items S_c between them need to be calculated, including the ratings for the common rated items of each initial reference user. This operation can be accomplished by comparing their respective rated items. The detailed algorithm about this operation is shown in Algorithm 4, where S_c represents the common rated items and S_{com} includes the complete rating information of each initial reference user to the common rated items. Afterwards, for all of the initial reference users, the calculation of PCC similarities between each of them and the target user can be achieved based on the ratings of their common rated items, as described in Equation (1) in Section 2.2. If the value of a PCC similarity is smaller than the designated threshold, then the corresponding initial user will be removed. Otherwise, the initial user will be grouped into the set of the final reference users.

By this two-step way of selecting reference users, we can select more reliable reference users of a target user and decrease the adverse effect caused by malicious users as possible. Meanwhile, this way can help alleviate the problem of cold start new users by means of acquiring reference information from social networks of users. Of course, we can only employ Step 1 to determine reference users if there is no sufficient historical ratings available. In addition, the threshold of PCC similarity in Step 2 can be adjusted to an appropriate value in terms of the prediction accuracy. In this sense, the way of determining the reference users of a target user is flexible in this study.

Algorithm 4: Calculate the common rated items S_c between the target user u_{tr} and each initial reference user, and complete rating information S_{com} of all reference users to S_c

Input: U_{iRef} , $M = (U_{id}, I_{id}, R)$, the target user u_{tr}
Output: the set S_c, S_{com}

```

1  $S_i = \Phi$ ,  $S_j = \Phi$ ,  $S_c = \Phi$ ,  $S_{com} = \Phi$ ;
2 for each element  $(u, i, r) \in M$  do
3   if  $u_{tr} == u$  then
4      $S_i = S_i \cup \{(u, i, r)\}$ ;
5   end
6   for each element  $u_j \in U_{iRef}$  do
7     if  $u_j == u$  then
8        $S_j = S_j \cup \{(u, i, r)\}$ ;
9     end
10  end
11 end
12 sort  $S_j$  according to user ID;
13 for each element  $(u_j, i_j, r_j) \in S_j$  do
14   for each element  $(u_i, i_i, r_i) \in S_i$  do
15     if  $i_j == i_i$  then
16        $S_c = S_c \cup u_j$ ;
17        $S_{com} = S_{com} \cup \{(u_j, i_j, r_j)\}$ ;
18     end
19   end
20 end
21 return  $S_c, S_{com}$ ;

```

4.3. Improving the Traditional CF Method Employing PCC Based on Users' Social Trust Network to Decrease Extreme Values

After we determine the reference users of a target user, the next work we confront is how to predict the unknown rating for the target user about the unknown target item—that is, how to aggregate the known ratings from the final reference users. As mentioned in Section 2.2, PCC is one of the widely used similarity metrics in a traditional CF method. Based on the results of the final reference users from a previous section, we will make an improvement for the traditional CF method employing PCC in this section based on users' social trust network in order to decrease extreme values during the prediction.

Suppose the number of the reference users of a target user u_i about an unknown item i_{un} is m . The average historical rating for each reference user u_j is represented as \bar{r}_j , where $j = 1, 2, \dots, m$. The average historical rating of the target user u_i is \bar{r}_i . The PCC similarity between each reference user u_j and the target user u_i is represented as $Sim(i, j)$. According to the traditional CF method, the rating prediction $\tilde{r}_{i, i_{un}}$ for the target user u_i to i_{un} is shown in the following Equation (3):

$$\tilde{r}_{i, i_{un}} = \bar{r}_i + \frac{\sum_{j=1}^m (r_j - \bar{r}_j) \times Sim(i, j)}{\sum_{j=1}^m Sim(i, j)} \quad (3)$$

where $Sim(i, j)$ is calculated according to Equation (1) in Section 2.2.

However, this prediction method has an obvious defect under the special circumstance. When all the reference users of the target user u_i have two kinds of opposite PCC similarities compared with u_i , for example, nearly half of similarities are very close to -1 and the rest of them are very close to 1 , in this situation, the denominator of Equation (3) will be a number close to 0 ; thus, the prediction result will be a very large number and become meaningless. Considering the extreme situation, we improve the Equation (3) of the traditional CF and transform it into the following Equation (4):

$$\tilde{r}_{i,un} = \bar{r}_i + \frac{\sum_{j=1}^m (r_j - \bar{r}_j) \times \text{Sim}(i, j)}{\sum_{j=1}^m |\text{Sim}(i, j)|} \quad (4)$$

Superficially, the improved Equation (4) can make prediction results smaller to some extent compared with Equation (3). In practice, it can make the traditional CF method employing PCC more robust, adaptable, and general by decreasing the number of extreme values, especially in large-scale samples' scenarios. The later experiments will demonstrate its effectiveness.

4.4. Introducing the LSI Factor into the Hybrid Recommendation Method

Up to now, we have determined the reference users of a target user by employing a trust relationship in users' social trust network, but the weight values of trust from different users have not been distinguished from one to another. In fact, for different reference users, the target user usually holds different trust degrees. However, it is very difficult for users themselves to distinguish the trust degrees among different users. In practice, it is also an important topic to determine trust weight in research domain of trust and reputation. In this section, we attempt to solve this problem by introducing LSI factor and describe its definition and calculation.

Based on the previous investigations in Section 2.4 and inspired by some work in social network, we believe that the social influence of different reference users plays different roles in decision-making of the same target user. Hence, we intend to introduce a local social influence factor LSI into our method and regard it as the personalized trust weight of a target user to different reference users. It can not only help realize the personalized trust for a target user to different reference users, but also help enhance the recommendation accuracy. Next, we introduce its definition and calculation, respectively. To describe clearly, we assume that there is only a direct trust relationship between the target user and each of the reference users in this section.

As described in Section 2.4, social influence is a quantitative measure and local social influence is a more efficient way in quantification because of its limited scope compared with global social influence. Liu et al. [53] consider social influence as a measure representing the degree for one user to affect the behavior of others and formalize local social influence as Equation (2). Inspired by their work, we define local social influence factor LSI and employ it into the users' social trust network to represent the trust weight of a target user to one of his reference users. As mentioned before, trust is a one-way relationship. This means that Bob trusts Alice does not represent that Alice also trusts Bob. Therefore, we need to differentiate a trust direction from one to another in users' social networks when defining local social influence.

Different from Liu et al. [53], the LSI factor in this study is redefined and employed to describe the influence of each reference user to a particular target user in their social trust network. For a certain reference user u_j , the LSI factor is determined by two parts: one is the number of trustors of the reference user, i.e., the number of users who trust u_j ; the other part is the total number of trustors from each reference user of the current target user u_i . The LSI factor of a reference user to the target user is defined as the ratio of the above two parts, as shown in Equation (5), supposing u_i has l reference users in total and each reference user u_j has $n_{tr}(u_j)$ trustors:

$$LSI(u_i \leftarrow u_j) = \frac{n_{tr}(u_j)}{\sum_{j=1}^l n_{tr}(u_j)}, \quad (j = 1, 2, \dots, l) \quad (5)$$

The first part shows the absolute importance of a reference user. That is, if one reference user is trusted by a large number of other users directly, then this user has great social influence and thus will be easily trusted by the target user. The second part aims to provide each reference user of the current target user with a common measure benchmark. In other words, it can be regarded as from the perspective of the target user to measure the importance of each reference user. Hence, the LSI factor of a reference user denotes relative importance of the reference user compared with all reference users

of the same target user. Based on the above definition and explanations, we can also regard the LSI factor as the trust degree of a target user to a certain reference user in their social trust network.

Let's take an example. As shown in Figure 3, the arrow represents the trust direction. Suppose the target user is T , who has four reference users i.e., trustees including A, B, C, D in their social trust network. Now let's calculate the respective LSI of A, B, C, D to T . For user A , from the Figure 3, we can see that A has four trustors in total. Similarly, users B, C, D , respectively, have 3, 3, 4 trustors. The total number of trustors of all reference users is $4 + 3 + 3 + 4 = 14$. According to the definition of LSI, their respective LSI value is $\frac{4}{14}, \frac{3}{14}, \frac{3}{14}, \frac{4}{14}$. Note that these LSI values are calculated for the target user T . In this sense, we consider the trust weight for target user T to A, B, C, D is respectively $\frac{4}{14}, \frac{3}{14}, \frac{3}{14}, \frac{4}{14}$. Apparently, LSI is a variable value to different target users even for the same reference user.

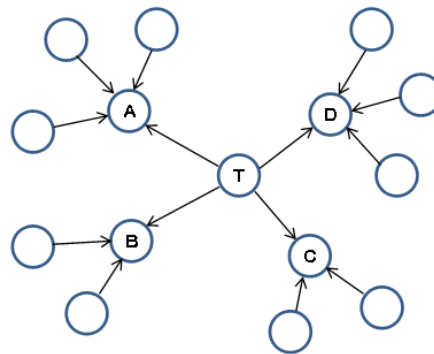


Figure 3. An illustration example of the proposed definition of LSI (Local Social Influence).

4.5. Predicting Ratings of Unknown Items for a Target User and Recommending Top-K Items

Based on the above work in Sections 4.3 and 4.4, now we can combine them into a new hybrid prediction method by adding an LSI factor in Equation (4), as shown in the following Equation (6):

$$\tilde{r}_{i,um} = \bar{r}_i + \frac{\sum_{j=1}^m (r_j - \bar{r}_j) \times \text{Sim}(i, j) \times \text{LSI}(u_i \leftarrow u_j)}{\sum_{j=1}^m |\text{Sim}(i, j)|} \quad (6)$$

According to Equation (6), we can predict an unknown rating for any target user about his unknown target item. Furthermore, we can achieve the task of predicting unknown ratings for a target user about multiple unknown target items employing the proposed hybrid method. Afterwards, we can make recommendations by providing the target user with the best or Top-k target items according to the prediction ratings.

Now, let's make a brief summary about the process of the proposed method in this section. At the beginning, we need to construct users' social trust networks according to the trust relationship from social networks. Next, one important work is to determine reference users by means of a two-step way of utilizing direct trust and setting the PCC threshold. After that, we need to calculate each LSI factor for each final reference user and then apply the improved traditional CF method integrating LSI factor to predict unknown ratings for the target user about multiple unknown items. Finally, we can make recommendations by providing the target user with Top-k items in terms of predicted ratings.

In conclusion, the proposed hybrid method can be utilized to predict multiple unknown items for a target user and acquire multiple predicted ratings accordingly. Thus, it can also be used to realize recommendations.

5. Experiments and Results

In this section, we first outline the experimental objectives and the dataset used in our experiments. Then, we conduct four sets of experiments to verify the proposed hybrid method based on a

well-known real dataset Epinions widely applied in social networks. Afterwards, we make detailed analyses employing two common accuracy metrics in recommendation systems based on the results of related experiments.

5.1. Experimental Objectives and Dataset Description

To demonstrate the feasibility, effectiveness and advantages of our proposed hybrid recommendation method, we design and conduct three sets of experiments. Experiment 1 is designed to verify the feasibility of the proposed hybrid method. Experiment 2 is to make comparisons between the traditional CF method and the improved CF method employing PCC based on the social trust network. Experiment 3 is to demonstrate the advantage of introducing LSI factor into the proposed method. Experiment 4 is to show the advantage of employing the two-step way of determining reference users of target users.

In the following experiments, we select the real dataset Epinions in a social network and download them from the website (http://www.trustlet.org/downloaded_epinions.html) as the data source of our experiments. The dataset was collected by Paolo Massa in a 5-week crawl (November/December 2003) from the Epinions.com website. It consists of two data files: ratings_data.txt and trust_data.txt. The former file is made up of triple tuples: user_id, item_id and rating_value, in which user_id is in range [1, 49,290], item_id is in range [1, 139,738], and rating_value is in range [1, 5]. In addition, the latter file is composed of trust relationship triple tuples shaped like source_user_id (i.e., trustor_id), target_user_id (i.e., trustee_id), trust_statement_value, among which the trust_statement_value only uses 1 to represent trust.

5.2. Preliminary Work

In this section, we first outline how to construct users' social trust networks based on an Epinions dataset, then describe how to determine reference users in later experiments, next depict our general experimental idea, and finally make necessary preparations for the following experiments.

To facilitate the descriptions of next experiments, we rename the two files from Epinions dataset to ratings.txt and trust.txt, respectively. Meanwhile, we delete the extra explanation information in these two files and improve the data format to be available in a Python programming environment. In addition, for the ease of constructing users' social trust network, we use 0 to denote no trust relationship between two users, as we have mentioned in Section 4.1. It is equivalent to complementing a zero value to trust_statement_value not included in trust.txt file, which will not have an effect to experimental results.

Since the trust.txt file derived from Epinions has provided us with an existing trust relationship, it is easy for us to construct users' social trust network $G = (U, T, W)$ according to Algorithm 1. In fact, the set U concerns all users whose ID ranges in [1, 49290]. T and W can also be acquired according to the trust.txt file. Considering that the LSI factor included in W is only used by a small amount of randomly selected target users, we take a simplified way to deal with it in later experiments. That is, we begin to calculate LSI only when we are ready to use it in later prediction.

Based on the social trust network, for a certain target user u_i who needs to be offered the rating prediction about an unknown item i_{un} , the preliminary job is to select the appropriate reference users for u_i . Considering that the Epinions dataset provides us with rich trust relationship, in the following experiments, we adopt a simplified way to search for the reference users of u_i . That is, determining reference users of a target user is only based on the direct trust relationship. Meanwhile, considering Leave One Out method mentioned later, to discover the reference users of a certain target user is thus equivalent to search for the users who satisfy the following conditions at the same time:

- (1) who have been trusted by the target user, i.e., direct trust users of the target user in their social trust network;
- (2) who have rated the target item that is unknown for the target user;

- (3) who have rated at least 3 items;
- (4) who have the similarities to the target user not smaller than the designated threshold of PCC.

In order to achieve our experimental objectives, we intend to utilize a Leave One Out evaluation method as mentioned by Liu et al. in their work [53] to perform all the experiments. The idea of a Leave One Out method is to hide one or more known ratings of a target user on purpose and then attempt to predict them successfully, aiming to compare between the prediction results and real ratings. In doing so, we first randomly determine one or more target users who have rated at least 3 items, then randomly select one item among those rated by target users as the unknown item to be predicted later. Next, we describe the idea to realize this process in Python programming.

For ease of use later, we first count rated items for each user in ratings.txt and keep the result in a list of counts for all users. Then, we randomly generates an integer ranging in [1, 49290], which is called user ID. In addition, we use this user ID as a position index to search the list of counts to see if the corresponding count is equal to or bigger than 3. If it is, then we select this user as a target user. If not, we repeat the process until we find one. After we determine the user ID of a target user, then we generate another random number which is not bigger than the item count that the target user has rated in order to determine the order of the rated item to be selected. By this way, now we can acquire the user ID of the selected target user u_i and the item ID which is regarded as the unknown item i_{un} to be predicted later.

In addition, we select all the users who have rated at least three items and save their related rating information to a new list; each element in the list is a triple tuple including user ID, rated item ID, and item rating. In fact, the total number of users who have rated at least three items is 28487.

In order to facilitate the following experiments, all related algorithms needed in experiments are realized in Python 3.7.0. The operating system environment is Microsoft Windows 7 (64 bit), Service Pack 1 (Microsoft Corporation, Seattle, WA, USA). The CPU is Intel (R) Core (TM) i5-4300U @ 1.90GHz 2.50GHz (Intel Corporation, Santa Clara, CA, USA). RAM is 8.00 GB (7.73 GB available).

5.3. Experiment 1: Verifying the Feasibility of the Proposed Method by an Example

In this experiment, we verify the proposed method by a simple example and perform the experiment in detail according to the procedures in Section 4. To attain this goal, we utilize the Leave One Out evaluation method mentioned above.

Now, let's take an example in detail. We randomly generate a target user u_i whose ID is 11155, and randomly select one of his rated item as an unknown item i_{un} to be predicted later, whose ID is 3588. We denote them as u_{11155} and i_{3588} separately. In fact, u_{11155} has rated 27 items and the detailed rating information is shown in Table 1. From the table, we can see that the real rating of item i_{3588} is 4, represented as $r_{i_{3588}} = 4$. In this experiment, we set the threshold of PCC to be 0.2. Next, we will employ our proposed hybrid method to predict the rating $\tilde{r}_{i_{3588}}$ of the item i_{3588} for the target user u_{11155} .

Table 1. Known ratings of a target user u_{11155} .

User ID	Item ID	Ratings	User ID	Item ID	Ratings	User ID	Item ID	Ratings	User ID	Item ID	Ratings
11155	10314	5	11155	19132	5	11155	363	3	11155	1072	4
11155	2211	5	11155	43653	1	11155	107931	5	11155	22779	5
11155	51497	4	11155	119161	5	11155	27354	5	11155	52966	5
11155	119162	3	11155	2738	4	11155	55152	4	11155	119163	5
11155	3004	5	11155	61521	2	11155	119164	5	11155	30804	5
11155	63078	5	11155	15707	5	11155	3560	5	11155	66469	4
11155	18889	3	11155	3588	4	11155	73308	4	-	-	-

To search for the reference users of u_{11155} , the first thing we need to do is to determine the initial reference users being trusted by u_{11155} according to the trust relationship in their social trust network.

Meanwhile, these users should have rated the item i_{3588} . In terms of the above conditions and Step 1 in Algorithm 2 of Section 4.2, we get the initial reference users of u_{11155} , including u_{10192} , u_{15569} , u_{20324} . Based on the Step 2 in Algorithm 2, we obtain their common rated items with u_{11155} and similarities to u_{11155} , which are listed in Table 2. Here, we can find that the minimum of the common rated items should be 2 in order to be able to calculate PCC similarity later. Since the threshold of PCC similarities is set to be 0.2, it is easy to see that the final reference users of u_{11155} are u_{10912} and u_{15569} .

Table 2. Determining reference users of the target user u_{11155} .

Target User	Initial Reference Users	Common Rated Items	Similarities	Final Reference Users
u_{11155}	u_{10912}	i_{3588}, i_{3004}	1.0	u_{10912}
	u_{15569}	$i_{27354}, i_{3004}, i_{3588},$	0.34028	u_{15569}
	-	$i_{363}, i_{51497}, i_{52966}$	-	-
	u_{20324}	$i_{19132}, i_{3004}, i_{3588}$	-1.0	-

Next, we need to calculate each LSI value of two reference users to the target user, i.e., $LSI(u_{11155} \leftarrow u_{10912})$ and $LSI(u_{11155} \leftarrow u_{15569})$. According to Equation (5), we can realize an algorithm to find all trustors of each reference user by traversing each line of the trust.txt file. Here, we obtain the count of each reference user, which is 12 and 25, respectively. Then, we can get $LSI(u_{11155} \leftarrow u_{10912}) = \frac{12}{12+25} = 0.32432$, $LSI(u_{11155} \leftarrow u_{15569}) = \frac{25}{12+25} = 0.67568$. In terms of Equation (4) in Section 4.3, we can further simplify the prediction rating formula of i_{3588} for the target user u_{11155} as the following Equation (7):

$$\tilde{r}_{i_{3588}} = \bar{r}_{11155} + \frac{\sum_{j=1}^2 (r_j - \bar{r}_j) \times Sim(11155, j) \times LSI(u_{11155} \leftarrow u_j)}{\sum_{j=1}^2 |Sim(i, j)|} \quad (7)$$

where we can calculate average results according to their known ratings: $\bar{r}_{11155} = 4.26923$ (Note: the rating $r_{i_{3588}}$ is not included), $\bar{r}_1 = 3.66667$, $\bar{r}_2 = 4.25$; while known ratings $r_1 = 2$ (i.e., the rating of u_{10912} to the target item i_{3588}), $r_2 = 4$ (i.e., the rating of u_{15569} to the target item i_{3588}), thus we can further get the final rating prediction for the target user, as shown in the following Equation (8):

$$\tilde{r}_{i_{3588}} = 4.26923 + \frac{(2 - 3.66667) \times 1.0 \times 0.32432 + (4 - 4.25) \times 0.34028 \times 0.67568}{1.0 + 0.34028} = 3.82304 \quad (8)$$

Compared with the real rating of u_{11155} , i.e., $r_{i_{3588}} = 4$, it is easy to see, the Mean Absolute Error i.e., $MAE = \frac{|3.82304 - 4|}{1} = 0.17696$.

If we take traditional CF method i.e., Equation (3) to predict the rating, for this example, we get the rating prediction value $\tilde{r}_{i_{3588}} = 2.96223$. Similarly, we can get the rating prediction using improved CF method. The comparison experiment results of three methods in which we take one step and two steps in Algorithm 2 separately to select reference users are shown as Table 3. For ease of description, from now on, we abbreviate traditional CF method as t-CF, improved CF method as i-CF, and the hybrid method combining LSI factor and improved CF method as h-CF.

Table 3. Rating prediction results of three methods for u_{11155} to i_{3588} .

Determining Reference Users of $u_{11,155}$		Step 1 (3 Reference Users)	Step 1 and Step 2 (2 Reference Users)	MAE (Step 1)	MAE (Step 1 and Step 2)
three methods	t-CF	-1.59353	2.96223	5.59353	1.03777
	i-CF	3.41678	3.82304	0.58322	0.17696
	h-CF	4.05314	3.82304	0.05314	0.17696

From the results in Table 3, we can find that the proposed hybrid method in this example achieves great improvement in rating prediction accuracy. However, the experiment with only one sample is far

from enough. MAE and RMSE (Rooted Mean Squared Error) are the same and simple in this example because we randomly take only one item as the target item to be predicted ratings. In later sections, we will conduct experiments with great amounts of random samples and define MAE and RMSE in this study to further verify the effectiveness of the proposed method.

After predicting multiple unknown ratings successfully by repeating the same process above, we can make recommendations by offering the Top-k rank items with higher ratings to the target user. Here, we won't describe that due to its ease of understanding.

So far, the example in this section accomplishes the goal of presenting the feasibility of the proposed method in rating prediction. In addition, it can offer subsequent help in making recommendations for target users.

5.4. Preparing Data for Experiment 2 and Experiment 3

Considering the occasional cases may often occur in experiments with a small size of samples when verifying experimental conclusions, we base our experiments on a certain amount of random samples. To achieve this goal, we first produce 100 groups of random target users. In each group, there is a common random item that is rated by each target user in this group. Afterwards, we should determine if all these groups meet the needs of the following experiments. If not, they should be omitted. The data preparation for later experiments are described as follows.

5.4.1. Determining Target Users in Each Group

In the previous section, our experiment is based on a single random target user and one of his random rated items. In this experiment, we will determine a lot of target users divided into 100 groups. For each group, we first randomly select one user from the list of those who have rated at least three items as the first target user and one of his rated items as the target item to be predicted by using the same way mentioned in Section 5.3. Then, we search the whole trust network for all users who have rated the same items as the first target user, and save them including related rating information in order to make rating predictions for all target users in this group about the same target item at a time.

For example, when we randomly determine the first target user u_{11155} and the target item i_{3588} , we can search the ratings.txt file for all the users who have rated item i_{3588} and save all the related information in a file. Each line in this result file has the same structure as ratings.txt and looks like "[11155, 3588, 4]". Repeat the same process as above until we generate 100 files i.e., 100 groups of target users, accompanied with 100 corresponding target items, and respective ratings from each of the target users in 100 groups. This means that each file may consist of multiple user IDs, the same target item ID, and different ratings to the same item.

After that, we get 100 random item IDs, respectively, from 100 group files, as listed in Table 4. Then, we check 100 corresponding group files and filter out 20 groups that consist of a few quantity of reference users from 1 to 4 in order to avoid the PCC's calculation failure later due to no common rated items. The 20 deleted target item IDs are shown as Table 5. The remaining 80 groups are kept for the next job.

Table 4. 100 random target item IDs, respectively, from 100 group files.

Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID
79527	12945	69398	9665	130033	17522	19631	52972	113856	42555	43470	137621	11054
27503	8560	4473	10367	119703	107834	99291	36492	1095	6475	25137	39524	43605
49406	11684	62666	42424	128463	113712	13710	5639	111016	33349	3706	13987	106055
10373	16432	5677	90856	2216	19679	57661	13491	30804	516	42464	135046	93933
498	49117	45042	11188	9655	4037	545	131303	46436	1539	3428	43653	113889
10870	24149	42251	3424	134506	43485	2655	138654	19717	128773	579	3036	615
2811	6131	44255	9114	66373	115311	19649	17053	104571	2458	129139	43385	393
11443	26966	16903	9689	71858	67602	43044	49586	33389	-	-	-	-

Table 5. 20 deleted target item IDs.

Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID
137621	99291	128463	130033	131303	46436	134506	138654	128773	135046
113889	42555	129139	119703	113712	111016	106055	90856	9689	104571

5.4.2. Determining Reference Users for Target Users in Each Group

According to our proposed method, determining reference users for each target user in one group need two steps as described in Section 4.2. In Step 1, we select the initial reference users from the trust network G for each target user in each group. In Step 2, we calculate PCC similarities between each reference user and the current target user and filter out the final reference users according to the predefined PCC similarity threshold.

In Step 1, when we attempt to search each initial reference user for each target user in a group, we need to traverse each line of trust.txt and meanwhile traverse the group file to check if the trustor in the current line is a target user in the current group. If so, then we take the trustee in the current line of trust.txt out and further check if the trustee has rated the same target item (i.e., to find if the trustee is also on the list of target users in the current group file). If also so, then save his corresponding user ID to a list of initial reference users of the current target user; if not, continue to traverse by the same way until to the end of any file: group file or trust.txt file. All lists of the initial reference users of each target user in the same group form a new list and are also saved in a new file.

Based on the above operation, we can get 80 files about the initial reference users matching 80 group files one by one. In order to guarantee the normal execution of next operation, we then check each of them to pick out the files in which include at least one target user whose list of the initial reference users is not empty. As a result, eight files have no initial reference users of any target user and thus they are deleted after this check, in which corresponding target item IDs concerns are listed in Table 6.

Table 6. Eight deleted target item IDs.

Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID	Item ID
57661	93933	67602	10870	113856	49586	115311	33389

In Step 2, we need to further filter reference users for all the target users according to the given threshold of PCC similarities. Before this, we should calculate PCC similarities between each target user and each of his reference users. In fact, before we compute PCC similarities, as mentioned in Experiment 1, we should check if the minimum of common rated items between each target user and each of his initial reference users is 2, and, meanwhile, the target item is included. If not, the value of PCC similarity can not be calculated, i.e., becomes meaningless. Based on this idea, we further filtering out one invalid file which includes the target item ID 71858. Now, we have 71 files for next calculation of PCC similarities and later ratings' prediction. For the ease of later experiments, we take a simplified way to carry out later experiments. In Experiment 2, we in essence adopt the default threshold -1 to determine the reference users. In Experiment 3, we just verify the role of the two-step way to determine reference users in rating prediction. Both of two experiments reveals the feasibility and effectiveness of the proposed method.

5.5. Experiment 2: Comparing Five Methods Based on PCC Similarity

In this section, we aim to make comparisons among five different methods based on PCC: the traditional CF method (t-CF), RTCF, f-PCC, the improved CF method (i-CF) and the hybrid recommendation method (h-CF), wherein the RTCF method is proposed by Parham Moradi et al. in Ref. [1] and f-PCC by Shuai Ding et al. in Ref. [33]. To achieve this goal, we attempt to conduct five sets of experiments based on the obtained 71 files in the previous section, separately employing five

methods based on PCC, i.e., Equation (3), Equation (9), Equation (10), Equation (4), and Equation (6). According to the descriptions in Section 5.4, obviously, we do not set the PCC similarity threshold for reference users in any group. This means that the values of PCC similarities vary in range $[-1, 1]$ in this section.

According to the RTCF method, when predicting an unknown target item for a target user who satisfies both conditions proposed in this study; the calculation formula is shown in Equation (9):

$$\tilde{r}_{i,i_{un}} = \bar{r}_i + \frac{\sum_{j=1}^m (r_j - \bar{r}_j) \times w_{i,j}(i, j)}{\sum_{j=1}^m w_{i,j}(i, j)} \quad (9)$$

where $w_{i,j}(i, j) = \frac{2 \times \text{Sim}(i, j) \times T_{i,j}}{\text{Sim}(i, j) + T_{i,j}}$. In fact, the trust value $T_{i,j}$ between user i and j is always regarded as 1 when we only consider direct trust relationship. The $\text{Sim}(i, j)$ can be calculated according to Equation (1).

Similarly, we can derive the prediction formula of the f-PCC adapted from Ref. [33], which can be used to predict an unknown rating for the target user, as shown in Equation (10):

$$\tilde{r}_{i,i_{un}} = \bar{r}_i + \frac{\sum_{j=1}^m (r_j - \bar{r}_j) \times \text{Sim}^C(i, j)}{\sum_{j=1}^m \text{Sim}^C(i, j)} \quad (10)$$

where $\text{Sim}^C(i, j) = C_{i,j} \times \text{Sim}(i, j)$, $C_{i,j} = 2^{\frac{|I_{i,j}|}{\max_{j \in [1, N], j \neq i} (|I_{i,j}|)}} - 1$, $|I_{i,j}|$ represents the number of common items rated by user i and j in this study. The $\text{Sim}(i, j)$ can also be calculated according to Equation (1).

After performing five sets of experiments, we correspondingly obtain five sets of results separately from 71 group files. To evaluate the prediction performance of these methods, we select MAE and RMSE metrics to make comparisons for the final prediction results. In doing so, we first compute MAE and RMSE for each group file, denoted as MAE_k and $RMSE_k$, where k means the k_{th} group file, and, here, $k = 1, 2, \dots, 71$. Suppose the number of valid target users in each group file is n_k , and the target item in k_{th} file is represented as i_k ; then, we can get the calculation formulas for MAE_k and $RMSE_k$ shown in Equation (11):

$$MAE_k = \frac{\sum_{j=1}^{n_k} |\tilde{r}_{j,i_k} - r_{j,i_k}|}{n_k} \quad (11)$$

$$RMSE_k = \sqrt{\frac{\sum_{j=1}^{n_k} (\tilde{r}_{j,i_k} - r_{j,i_k})^2}{n_k}}$$

We then separately compute the average values of MAE and RMSE for all group files, which are represented as \overline{MAE} and \overline{RMSE} . In addition, they are separately defined according to the following Equation (12). Here, n_f represents the number of valid files, which is equal to the number of target items to be predicted:

$$\overline{MAE} = \frac{\sum_{k=1}^{n_f} MAE_k}{n_f} \quad (12)$$

$$\overline{RMSE} = \frac{\sum_{k=1}^{n_f} RMSE_k}{n_f}$$

As mentioned in Section 4.3, the traditional CF method employing PCC similarity may generate extreme values in the process of rating predictions under some circumstances. Among the first set of experiments employing the traditional CF method, we find that the prediction results include three groups with extreme large values (the target item IDs are separately 16903, 30804, 393) and eight groups with second larger values (the target item IDs are separately 12945, 11054, 8560, 10367, 39524,

13710, 3424, and 615). For ease of description, we first make analyses based on the remaining 60 group results, and then make separate analyses about the 11 group results with extreme values.

Excluding another two groups (the corresponding target item IDs are 42424 and 13491 separately) for which the RTCF method obtains invalid results, the results of *MAE* and *RMSE* employing Equation (11) for five methods based on the same 58 group files are separately shown in Figures 4 and 5. The average values of *MAE* and *RMSE* in terms of Equation (12) for the 58 groups are shown in Figure 6. We can see that the hybrid method acquires the best performance in most cases in prediction accuracy among five methods, and the improved CF method can generate smaller errors and maintain a more steady tendency than the traditional CF method in most cases, especially in *RMSE*. The comparisons of *MAE* and *RMSE* in Figure 6 can verify the conclusions drawn above from the perspective of average level.

For the remaining 11 files, we respectively calculate the average values for *MAE* and *RMSE* according to Equation (11). Due to the remaining 11 group results with extreme values caused by the traditional CF method employing PCC, it is difficult to make graphs for them, so we take the form of a table to make comparisons. The comparison results of *MAE* and *RMSE* for five methods based on the rest 11 groups are presented in Table 7. Apparently, it is easy to find that the hybrid method can achieve the best performance in prediction accuracy compared with the other four methods on the whole, especially in the *RMSE* metric. We can also see that the improved CF can overcome the weakness of the traditional CF method and greatly reduce the errors to a limited and acceptable range.

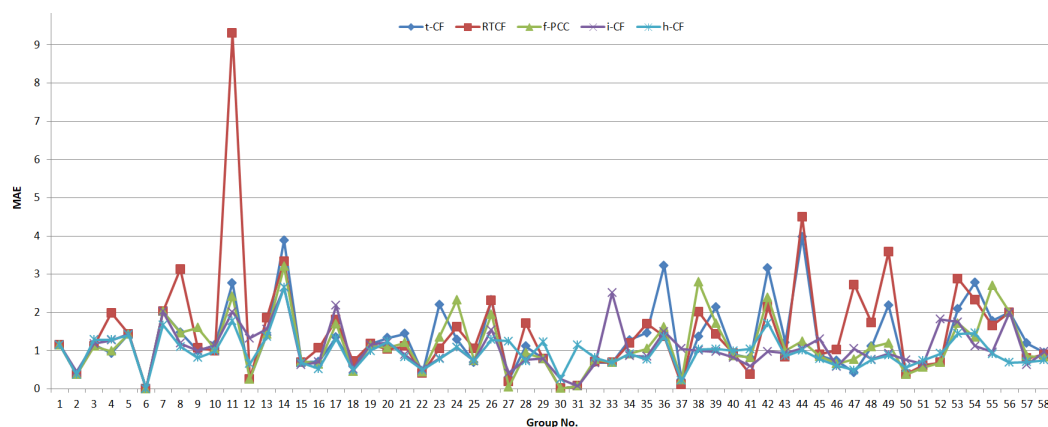


Figure 4. MAE (Mean Absolute Error) comparisons among the five methods based on the same 58 group target users.

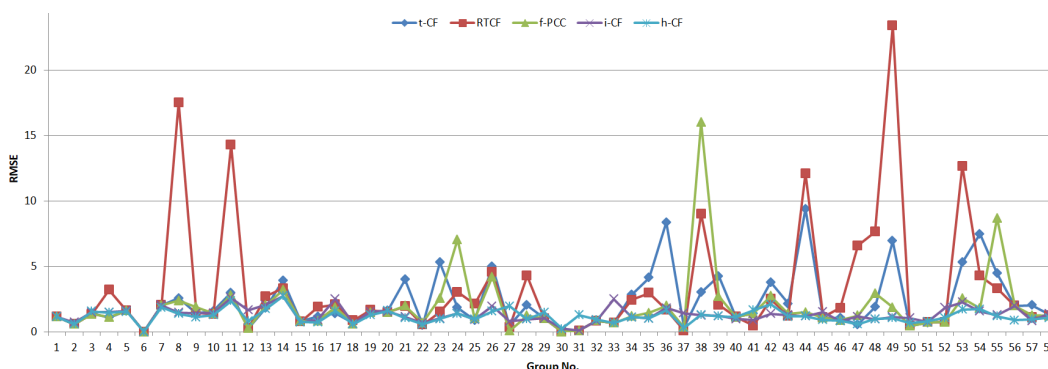


Figure 5. RMSE (Rooted Mean Squared Error) comparisons among the five methods based on the same 58 group target users.

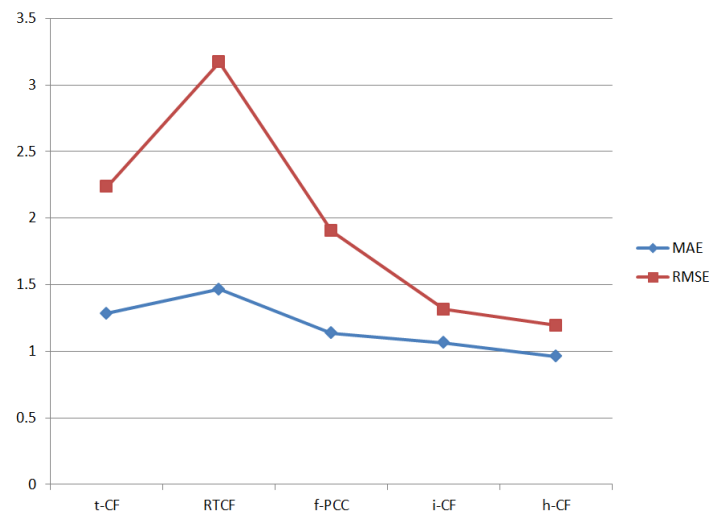


Figure 6. Average values of MAE and RMSE in five methods based on the same 58 group target users.

In terms of Equation (12), we can acquire the average value of MAE and $RMSE$, respectively, for the 11 groups. The values of \overline{MAE} and \overline{RMSE} for the 11 groups are shown in Table 8. Based on these results, we can also draw the conclusion that the proposed hybrid method can attain best accuracy performance than the other four CF methods based on PCC in extreme cases. It is worth noting that the proposed hybrid method outperforms the improved CF method mainly owing to the introduction of the LSI factor.

In addition, from Tables 7 and 8, we can also conclude that the h-CF is able to gain higher item coverage.

Table 7. Comparison among five methods based on 11 groups.

Target Item ID		16903	30804	393	12945	11054	8560	10367	39524	13710	3424	615
MAE	t-CF	4.5185×10^{12}	2.4474×10^{13}	9.5802×10^{13}	5.3191	32.8691	2.2830	4.3884	74.3301	7.2711	8.7180	2.5399
	RTCF	3.8074	1.3937	2.9811	0.3867	0.2695	0.6729	4.8870	1.4282	1.8584	2.0649	0.9619
	f-PCC	13.6699	1.7051	9.5802×10^{13}	0.5256	0.2257	0.8897	3.1888	2.1848	4.7662	1.9526	2.7093
	i-CF	1.2367	0.9235	0.9218	0.3882	0.5212	0.7065	1.6819	1.3543	1.7803	1.9576	0.7446
	h-CF	1.1298	0.7035	0.8562	0.5429	0.7038	0.7738	1.6368	0.9223	1.2976	1.8972	0.7098
RMSE	t-CF	6.1125×10^{13}	1.5284×10^{14}	1.2380×10^{15}	16.8291	79.7243	21.5924	15.0557	216.2004	46.9569	34.6527	17.8772
	RTCF	28.8753	3.1751	16.5549	0.4838	0.3155	1.4548	28.5127	1.5474	3.6783	2.9647	1.6135
	f-PCC	128.8012	4.2697	1.2380×10^{15}	0.7267	0.2996	2.9838	11.7476	2.6660	23.8013	2.4989	26.6749
	i-CF	1.6713	1.2122	1.2067	0.4855	0.6295	0.9422	2.0937	1.5415	2.3622	2.3501	0.9620
	h-CF	1.4649	0.8891	1.1189	0.8008	0.7978	0.9073	1.9691	1.1039	1.6300	2.2378	0.8884

Table 8. Comparison between five methods.

Methods	Number of predicted groups	\overline{MAE}	\overline{RMSE}
t-CF	60 groups	1.26927	2.19003
RTCF	58 groups	1.46636	3.17218
f-PCC	60 groups	1.12778	1.86669
i-CF	60 groups	1.09305	1.33353
h-CF	60 groups	0.95963	1.18509
t-CF	11 groups	1.13450×10^{13}	1.32000×10^{14}
RTCF	11 groups	1.88256	8.10691
f-PCC	11 groups	8.70925×10^{12}	1.12548×10^{14}
i-CF	11 groups	1.11062	1.40519
h-CF	11 groups	1.01579	1.25526

5.6. Experiment 3: Demonstrating the Advantage of Determining Reference Users Employing the Proposed Two-Step Way

In this set of experiments, we attempt to demonstrate the advantage of determining reference users of a target user shown in Section 4.2. To achieve this goal, we conduct our experiments based on 71 groups of random target users with the same random target item in each group. Next, we will depict the experiment process and make analyses based on the results.

In order to identify the effects of different thresholds of PCC similarity on the prediction results, we perform several sets of experiments based on the traditional CF method employing PCC and respectively set the threshold of PCC similarity to be equal 0.0, 0.2, 0.4, and 0.6. Afterwards, we make comparisons according to the average values of MAE and $RMSE$ between each set of these results and the set of experimental results in Experiment 3 without any threshold based on the traditional CF employing PCC. The experimental results are shown in Table 9.

Table 9. Comparison for different thresholds of PCC similarity.

Threshold	None	0.0	0.2	0.4	0.6
number of valid groups	71	66	66	64	62
the range of PCC	$[-1,1]$	$[0,1]$	$[0.2,1]$	$[0.4,1]$	$[0.6,1]$
\overline{MAE}	1.75767×10^{12}	0.94049	0.94041	0.93650	0.99650
\overline{RMSE}	2.04507×10^{13}	1.16249	1.16355	1.15795	1.23190

From the results in Table 9, we can see that the prediction accuracy varies with the change of the PCC similarity threshold. At first, when the threshold increases, the average errors of MAE and $RMSE$ decrease greatly, respectively. When the PCC threshold is set at 0.4, the average errors reach the smallest amounts. However, as the PCC threshold continues to increase, the average errors begin to increase. This fluctuation implies that there exists an optimal threshold value of PCC for a set of the same samples. It also reveals that the two-step way of determining reference users can help improve the prediction accuracy to some extent. In addition, we can also observe that the number of valid groups decreases with the increase of PCC threshold. This reveals that the amount of the reference users of target users decreases with the enhanced condition of PCC similarity, which will affect the prediction accuracy to some extent.

As is observed from Experiment 2, the hybrid method is based on the traditional CF method and its performance has become better than CF; we can conclude that the hybrid method can obey the same rule on the influence of the threshold in a two-step way of determining reference users. However, the appropriate threshold depends on the real application scenarios.

6. Discussion

Overall, our proposed hybrid recommendation method is able to greatly improve the traditional CF method especially in accuracy compared with the other four methods. On the one hand, it can reduce the appearance of extreme prediction values under most circumstances by adapting the traditional CF method. From Figures 4 and 5, we can see that the h-CF is able to keep a relative steady state with lower errors compared with other methods. In addition, from Figure 6, we conclude that the proposed method can obtain the best accuracy at the average level. The result shown in Tables 7 and 8 can also reveal that the proposed method h-CF is able to acquire higher item coverage especially compared with t-CF and RTCF, which largely depends on the decreasing appearance of extreme values. On the other hand, the introduction of LSI derived from the social networks can not only further improve the accuracy but also distinguish the different trust degree among reference users from the perspective of target users, which can also make the approach more explainable. Moreover, Experiment 3 shows that the two-step way of determining reference users can help to identify appropriate reference users to improve prediction accuracy to some extent.

Although some trust-aware recommendation systems have been proposed, how to extract and refine trust relationship from the social networks in order to build the trust network is seldom taken into account. In this study, we present the algorithm to build trust networks employing users' social networks, but we base our experiments on the existing trust relationship from the Epinions dataset. Considering the balance between the performance and complexity, our proposed method only employing direct trust is straightforward. In fact, indirect trust often plays a minor role in decision-making. Although some research work focuses on the trust propagation and application [1,3], whether the combination of indirect trust into the method can make better accuracy is unknown. The balance between the performance and complexity is still an important concern. In addition, the update of the LSI values concerns the dynamic iteration of the trust network. In addition, the relationship between the accuracy and the number of reference users also needs deep investigation.

7. Conclusions and Future Work

The CF-based recommendation approach has gained more attention in different fields. Except its outstanding advantages over the content-based recommendation, several weaknesses greatly set back the prediction accuracy of recommendation systems. To overcome the weaknesses and improve the prediction accuracy, we have proposed a hybrid collaborative filtering recommendation method incorporating direct trust of the target user and local social influence of reference users based on the users' social network network, which is derived from their social networks in this study. Several sets of experiments on the real dataset Epinions in social networks have shown the feasibility and effectiveness of the proposed hybrid recommendation method, especially noteworthy in enhancing the prediction accuracy.

Our main work is different from existing work in several ways. First, the direct trust relationships in the social trust network have been combined into the traditional CF method employing PCC similarity to ensure the reliability of rating data source. Therefore, the proposed method can decrease the effect of malicious behaviors and untrustworthy users. Second, a two-step way of determining reference users is presented by employing direct trust between users in social trust networks and PCC similarity thresholds, which can significantly improve the prediction accuracy when the optimal threshold is set. Third, the CF-based recommendation method employing PCC similarity has been improved by decreasing extreme values. The improvement can increase the robustness of the traditional CF method. Most important of all, the local social influence is introduced into the improved CF-based method that further increases the prediction accuracy and make the trust more personalized and explainable.

However, the proposed hybrid in this study still has several challenges to confront in the future. Since the proposed method utilizes a two-step way to determine the final reference users of a target user, one challenge is the quantity of reference users. Searching for more trustworthy reference

users by means of depth-first search and width-first search in the users' social trust network may be a good solution. Under such a circumstance, the proposed method have to be further extended. Another challenging work is the balance between the prediction accuracy and the threshold of PCC. This issue concerns the density of trusted reference users. Last but not least, how to apply the proposed recommendation method in various distributed service environments deserves deep investigation in our future work.

Author Contributions: The original idea of the proposed method, algorithms, the realization of the experiments, and the original draft were completed by L.L. Y.Y. supervised the whole work. X.C. finished review and editing. Z.L. attended the discussion of the idea and gave some advice. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China Grant No. 91118002 and the Natural Science Foundation of Hebei Province, China Grant No. F2019105134.

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

References

1. Moradi, P.; Ahmadian, S. A reliability-based recommendation method to improve trust-aware recommender systems. *Expert Syst. Appl.* **2015**, *42*, 7386–7398. [\[CrossRef\]](#)
2. Liu, H.; Hu, Z.; Mian, A.; Tian, H.; Zhu, X. A new user similarity model to improve the accuracy of collaborative filtering. *Knowl. Based Syst.* **2014**, *56*, 156–166. [\[CrossRef\]](#)
3. Chen, C.C.; Wan, Y.H.; Chung, M.C.; Sun, Y.C. An effective recommendation method for cold start new users using trust and distrust networks. *Inf. Sci.* **2013**, *224*, 19–36. [\[CrossRef\]](#)
4. Parvin, H.; Moradi, P.; Esmaeili, S. TCFACO: Trust-aware collaborative filtering method based on ant colony optimization. *Expert Syst. Appl.* **2019**, *118*, 152–168. [\[CrossRef\]](#)
5. Ahmadian, S.; Afsharchi, M.; Meghdadi, M. An effective social recommendation method based on user reputation model and rating profile enhancement. *J. Inf. Sci.* **2019**, *45*, 607–642. [\[CrossRef\]](#)
6. Massa, P.; Avesani, P. Trust-Aware Collaborative Filtering for Recommender Systems. In *On the Move to Meaningful Internet Systems 2004: CoopIS, DOA, and ODBASE*; Meersman, R., Tari, Z., Eds.; Springer: Berlin/Heidelberg, Germany, 2004; pp. 492–508.
7. Bedi, P.; Sharma, R. Trust based recommender system using ant colony for trust computation. *Expert Syst. Appl.* **2012**, *39*, 1183–1190. [\[CrossRef\]](#)
8. Burke, R. Hybrid Recommender Systems: Survey and Experiments. *User Model. User Adapt. Interact.* **2002**, *12*, 331–370. [\[CrossRef\]](#)
9. Javari, A.; Jalili, M. Cluster-Based Collaborative Filtering for Sign Prediction in Social Networks with Positive and Negative Links. *ACM Trans. Intell. Syst. Technol.* **2014**, *5*, 1–19. [\[CrossRef\]](#)
10. Braidia, F.; Mello, C.E.; Pasinato, M.B.; Zimbardo, G. Transforming collaborative filtering into supervised learning. *Expert Syst. Appl.* **2015**, *42*, 4733–4742. [\[CrossRef\]](#)
11. Wei, J.; He, J.; Chen, K.; Zhou, Y.; Tang, Z. Collaborative filtering and deep learning based recommendation system for cold start items. *Expert Syst. Appl.* **2017**, *69*, 29–39. [\[CrossRef\]](#)
12. Davoudi, A.; Chatterjee, M. Social trust model for rating prediction in recommender systems: Effects of similarity, centrality, and social ties. *Online Soc. Netw. Media* **2018**, *7*, 1–11. [\[CrossRef\]](#)
13. Sun, Z.; Han, L.; Huang, W.; Wang, X.; Zeng, X.; Min, W.; Hong, Y. Recommender systems based on social networks. *J. Syst. Softw.* **2015**, *99*, 109–119. [\[CrossRef\]](#)
14. Nilashi, M.; Ibrahim, O.; Bagherifard, K. A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques. *Expert Syst. Appl.* **2018**, *92*, 507–520. [\[CrossRef\]](#)
15. Lee, W.P.; Ma, C.Y. Enhancing collaborative recommendation performance by combining user preference and trust-distrust propagation in social networks. *Knowl. Based Syst.* **2016**, *106*, 125–134. [\[CrossRef\]](#)
16. Sarker, I.H. Context-aware rule learning from smartphone data: Survey, challenges and future directions. *J. Big Data* **2019**, *6*, 95. [\[CrossRef\]](#)
17. Sarker, I.H.; Kayes, A.S.M.; Watters, P. Effectiveness analysis of machine learning classification models for predicting personalized context-aware smartphone usage. *J. Big Data* **2019**, *6*, 57. [\[CrossRef\]](#)

18. Sarker, I.H.; Kayes, A. ABC-RuleMiner: User behavioral rule-based machine learning method for context-aware intelligent services. *J. Netw. Comput. Appl.* **2020**, *168*, 102762. [\[CrossRef\]](#)
19. McAuley, J.; Leskovec, J. Hidden factors and hidden topics: Understanding rating dimensions with review text. In Proceedings of the RecSys 2013 7th ACM Conference on Recommender Systems, Como, Italy, 27–31 August 2013; pp. 165–172.
20. Sarker, I.H.; Colman, A.; Han, J. RecencyMiner: Mining recency-based personalized behavior from contextual smartphone data. *J. Big Data* **2019**, *6*, 49. [\[CrossRef\]](#)
21. Cheng, Z.; Ding, Y.; He, X.; Zhu, L.; Song, X.; Kankanhalli, M. A3NCF: An adaptive aspect attention model for rating prediction. In Proceedings of the IJCAI International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 3–19 July 2018; pp. 3748–3754.
22. Cheng, Z.; Chang, X.; Zhu, L.; Kanjirathinkal, R.C.; Kankanhalli, M. MMalfM: Explainable recommendation by leveraging reviews and images. *ACM Trans. Inf. Syst.* **2019**, *37*, 1–28. [\[CrossRef\]](#)
23. Yang, J.H.; Wang, C.J.; Chen, C.M.; Tsai, M.F. HoP-Rec: High-order proximity for implicit recommendation. In Proceedings of the RecSys 2018—12th ACM Conference on Recommender Systems, Vancouver, BC, Canada, 2–7 October 2018; pp. 140–144.
24. Chen, C.M.; Tsai, M.F.; Wang, C.J.; Yang, Y.H. Collaborative similarity embedding for recommender systems. In Proceedings of the The Web Conference 2019—Proceedings of the World Wide Web Conference (WWW 2019), San Francisco, CA, USA, 13–17 May 2019; pp. 2637–2643.
25. Massa, P.; Avesani, P. Trust-Aware Recommender Systems. In Proceedings of the 2007 ACM Conference on Recommender Systems, Minneapolis, MN, USA, 19–20 October 2007; Association for Computing Machinery: New York, NY, USA, 2007; pp. 17–24. [\[CrossRef\]](#)
26. Cheng, L.C.; Wang, H.A. A fuzzy recommender system based on the integration of subjective preferences and objective information. *Appl. Soft Comput.* **2014**, *18*, 290–301. [\[CrossRef\]](#)
27. Koohi, H.; Kiani, K. A new method to find neighbor users that improves the performance of Collaborative Filtering. *Expert Syst. Appl.* **2017**, *83*, 30–39. [\[CrossRef\]](#)
28. Bobadilla, J.; Ortega, F.; Hernando, A. Recommender systems survey. *Knowl. Based Syst.* **2013**, *46*, 109–132. [\[CrossRef\]](#)
29. Aggarwal, C.C. *Recommender Systems*; Springer International Publishing: Cham, Switzerland, 2016. [\[CrossRef\]](#)
30. Jannach, D.; Zanker, M.; Felfernig, A.; Friedrich, G. *Recommender Systems: An Introduction*; Cambridge University Press: Cambridge, UK, 2010. [\[CrossRef\]](#)
31. Nilashi, M.; Ibrahim, O.B.; Ithnin, N. Hybrid recommendation approaches for multi-criteria collaborative filtering. *Expert Syst. Appl.* **2014**, *41*, 3879–3900. [\[CrossRef\]](#)
32. Boratto, L.; Carta, S. Using Collaborative Filtering to Overcome the Curse of Dimensionality when Clustering Users in a Group Recommender System. In Proceedings of the International Conference on Enterprise Information Systems, Lisbon, Portugal, 27–30 April 2014; pp. 564–572. [\[CrossRef\]](#)
33. Shuai, D.; Yang, S.; Zhang, Y.; Liang, C.; Xia, C. Combining QoS prediction and customer satisfaction estimation to solve cloud service trustworthiness evaluation problems. *Knowl. Based Syst.* **2014**, *56*, 216–225.
34. Zhijun, Z.; Hong, L. Social recommendation model combining trust propagation and sequential behaviors. *Appl. Intell.* **2015**, *43*, 695–706. [\[CrossRef\]](#)
35. Lu, L.; Yuan, Y. A Novel TOPSIS Evaluation Scheme for Cloud Service Trustworthiness Combining Objective and Subjective Aspects. *J. Syst. Softw.* **2018**, *143*, 71–86. [\[CrossRef\]](#)
36. Cho, J.H.; Swami, A.; Chen, I.R. A Survey on Trust Management for Mobile Ad Hoc Networks. *IEEE Commun. Surv. Tutor.* **2010**, *2*, 562–583. [\[CrossRef\]](#)
37. Chen, X.; Yuan, Y.; Orgun, M.A. Using Bayesian networks with hidden variables for identifying trustworthy users in social networks. *J. Inf. Sci.* **2019**, 1–16. [\[CrossRef\]](#)
38. Hussain, F.K.; Chang, E. An Overview of the Interpretations of Trust and Reputation. In Proceedings of the Advanced International Conference on Telecommunications, Glasgow, UK, 24–28 June 2007; pp. 826–830. [\[CrossRef\]](#)
39. Adali, S. *Trust as a Computational Concept*; Springer: New York, NY, USA, 2013; pp. 5–24.
40. Gambetta, D. Can we trust trust? *Trust Mak. Break. Coop. Relat.* **1988**, *5*, 213–237.
41. Jø Sang, A.; Presti, S.H.L. Analysing the Relationship between Risk and Trust. *Lect. Notes Comput. Sci.* **2004**, *lncs 2*, 135–145.

42. Ziegler, C.N.; Lausen, G. Propagation Models for Trust and Distrust in Social Networks. *Inf. Syst. Front.* **2005**, *7*, 337–358. [\[CrossRef\]](#)
43. Chen, W.K. *Linear Networks and Systems-Algorithms and Computer-Aided Implementations*; World Scientific: London, UK, 1994.
44. Mui, L.; Mohtashemi, M.; Halberstadt, A. A Computational Model of Trust and Reputation for E-businesses. In Proceedings of the Hawaii International Conference on System Sciences, Big Island, HI, USA, 7–10 January 2002; pp. 2431–2439.
45. Wu, F.; Li, H.H.; Kuo, Y.H. Reputation evaluation for choosing a trustworthy counterparty in C2C e-commerce. *Electron. Commer. Res. Appl.* **2011**, *10*, 428–436. [\[CrossRef\]](#)
46. Yang, X.; Guo, Y.; Liu, Y.; Steck, H. A survey of collaborative filtering based social recommender systems. *Comput. Commun.* **2014**, *41*, 1–10. [\[CrossRef\]](#)
47. Seo, Y.D.; Kim, Y.G.; Lee, E.; Baik, D.K. Personalized recommender system based on friendship strength in social network services. *Expert Syst. Appl.* **2017**, *69*, 135–148. [\[CrossRef\]](#)
48. Zheng, X.L.; Chen, C.C.; Hung, J.L.; He, W.; Hong, F.X.; Lin, Z. A Hybrid Trust-Based Recommender System for Online Communities of Practice. *IEEE Trans. Learn. Technol.* **2015**, *8*, 345–356. [\[CrossRef\]](#)
49. Ye, M.; Liu, X.; Lee, W.C. Exploring Social Influence for Recommendation: A Generative Model Approach. In Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '12, Gold Coast, Australia, 6–11 July 2012; ACM: New York, NY, USA, 2012; pp. 671–680. [\[CrossRef\]](#)
50. Ma, H.; Zhou, T.C.; Lyu, M.R.; King, I. Improving Recommender Systems by Incorporating Social Contextual Information. *ACM Trans. Inf. Syst.* **2011**, *29*. [\[CrossRef\]](#)
51. Ye, M.; Yin, P.; Lee, W.C.; Lee, D.L. Exploiting Geographical Influence for Collaborative Point-of-interest Recommendation. In Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '11, Beijing, China, 25–29 July 2011; ACM: New York, NY, USA, 2011; pp. 325–334. [\[CrossRef\]](#)
52. Shang, S.; Hui, P.; Kulkarni, S.R.; Cuff, P.W. Wisdom of the Crowd: Incorporating Social Influence in Recommendation Models. In Proceedings of the 2011 IEEE 17th International Conference on Parallel and Distributed Systems, Tainan, Taiwan, 7–9 December 2011; pp. 835–840. [\[CrossRef\]](#)
53. Liu, H.; Hu, Z.; Tian, H.; Zhou, D. An adaptive social influence propagation model based on local network topology. *Lect. Notes Bus. Inf. Process.* **2013**, *152*, 14–26. [\[CrossRef\]](#)
54. Guha, R.; Kumar, R.; Raghavan, P.; Tomkins, A. Propagation of trust and distrust. In Proceedings of the International Conference on World Wide Web, New York, NY, USA, 17–20 May 2004; pp. 403–412.



© 2020 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/>).