CSCanada
Management Science and Engineering
Vol. 11, No. 3, 2017, pp. 9-19
DOI:10.3968/10046

ISSN 1913-0341 [Print]
ISSN 1913-035X [Online]
www.cscanada.net
www.cscanada.org

A Trust-Based Collaborative Filtering Algorithm Using a User Preference Clustering

SUN Nannan[a],*; ZHUANG Yayun[a]; NI Shuqing[a]

[a]Department of Management Science and Engineering, South China University of Technology, Guangzhou, China.
*Corresponding author.

Received 24 July 2017; accepted 11 September 2017
Published online 16 September 2017

Abstract

Collaborative filtering is a widely adopted approach to recommendation, but sparse and high dimensional data are often barriers to providing high quality recommendations. Meanwhile, the traditional methods only utilize the information of the user-item rating matrix but ignore the trust relations between users, so their recommendation precision is often unsatisfactory. To address such issues, this paper constructs an user-preference matrix to reduce the data dimension and clusters the users by *k*-means clustering algorithm. Incorporating trust relationship, an improved similarity method is proposed to compute the similarity value. Then we find the nearest neighbor in the target user's category according to the similarity; and predict the user's prediction score by the nearest neighbor. At last we recommend the items with high prediction score to the user. This improved method has been tested via MovieLens 100K in order to make a comparison with the traditional techniques. The results have indicated that the proposed method can enhance performance of recommender systems.

Key words: User preference clustering; Trust relationship; Collaborative filtering

Sun, N. N., Zhuang, Y. Y., & Ni, S. Q. (2017). A Trust-Based Collaborative Filtering Algorithm Using a User Preference Clustering. *Management Science and Engineering*, *11*(3), 9-19. Available from: URL: http://www.cscanada.net/index.php/mse/article/view/10046
DOI: http://dx.doi.org/10.3968/10046

INTRODUCTION

Collaborative filtering is one of the most successful recommendation technologies (Adomavicius & Tuzhilin, 2005) and has been widely used in various related fields of e-commerce recommendation system and Internet. In this approach, opinions and actions of other users with similar tastes (i.e. neighbor users) are used to produce suitable recommendations for a given user (i.e. an active user) in the recommendation process (Zhang & Zhou, 2012).This technique is an effective way to solve the problem of information overload. The advantage of this technique is that it is recommended personalization, not limited by the specific content of recommended items and has high recommendation reliability (Li & Yan, 2009). However, with the expansion of e-commerce system, the recommendation based on collaborative filtering is very time-consuming to search the nearest neighbor of the target user or target items, which greatly affects the system efficiency. At the same time, collaborative filtering faces the problem of sparse data (Park & Pennock, 2007). Taking the e-commerce recommendation system as an example, the total amount of products purchased by users in these systems usually only accounts for about 1% of the total amount of goods (Castro-Schez & Miguel, 2011)). User-item scoring data is extremely sparse, resulting in a sharp drop in the recommendation quality based on traditional similarity (Breese, Heckerman, & Kadie, 1998). In addition, the traditional collaborative filtering algorithm only considers the user's similarity of the ratings, without considering the influence of the users' trust relationship with the user's neighbor selection, which affects the accuracy of the recommendation.

In this paper, we propose a collaborative filtering model based on users' preference clustering and improved similarity. Based on the user-preference matrix, the algorithm clusters the users by *k*-means clustering algorithm, and divides the users with similar preference into the same cluster which can reduce the search space of the neighbor users. In addition, the model introduces trust relationship into the similarity measure, so that the target user can get a better neighbor set and get a better recommendation result.

1. RELATED WORK

1.1 User Based Collaborative Filtering Algorithms

Collaborative filtering analyzes the user-item relationship through user-item rating data and then uses this correlation to generate personalized recommendations for users based on the assumption that users with similar preferences may be interested in similar products. Traditional collaborative filtering algorithms generally can be classified into two categories: user-based collaborative filtering and item-based collaborative filtering. The former assumes that if two users had similar preferences for the item in the past, they still have a similar preference for the item now; the latter assumes that if a user used to like an item, the user still likes the similar item. We use user-based collaborative filtering algorithm as the basic algorithm in this paper.

Define the user evaluation matrix $R_{m\times n}$ is a matrix of m rows and n columns, where m is the number of users and n is the number of items. The user vector, $u_i \in U=\{u_1,u_2,\ldots,u_m\}m$ is the number of users, and U is the set of users. The item vector $i_j \in I=\{i_1,i_2,\ldots,i_n\}$, n is the number of items and I is the set of items in the system. $r_{i,j} \in R_{m\times n}$, $R_{i,j}$ denotes the score of user u_i on item i_j. If user u_i does not evaluate item i_j, then $r_{i,j}=0$. Otherwise, it is a non-zero rating value.

Collaborative filtering algorithm is trying to predict the u_i score of non-evaluation items i_j, which is $R_{i,j}$.u_i as the target user, i_j is called the target item. Collaborative filtering algorithm deals with the user-item matrix. The traditional collaborative filtering algorithm (CF) consists of two processes:

i) Finding the similar users of the target user through the similarity calculation;
ii) Generating a user's prediction score of the target item based on the score of nearest users, and selecting the user with the highest prediction score.

Traditional similarity calculation methods include: cosine similarity, correlation similarity.

(a) Cosine similarity calculation formula

$$\mathrm{sim}(u,v)=\cos(\vec{u},\vec{v})=\frac{\vec{u}\cdot\vec{v}}{||\vec{u}||^2 * ||\vec{v}||^2}. \quad (1)$$

u and v are considered as two scoring vectors in m-dimensional user space, and the similarity is calculated by calculating the cosine of the angle between two vectors.

(b) Correlation similarity calculation formula

$$\mathrm{sim}(u,v)=\frac{\sum_{i\in I_{u,v}}(R_{u,i}-\bar{U})(R_{u,i}-\bar{V})}{\sqrt{\sum_{i\in I_{u,v}}(R_{u,i}-\bar{U})^2}\sqrt{\sum_{i\in I_{u,v}}(R_{u,i}-\bar{V})^2}}. \quad (2)$$

$\bar{U}$ and $\bar{V}$ represent the average score of items scored by users u and v in their common scoring item set.

Another key step of the collaborative filtering algorithm is to generate the user's prediction score of the target item according to the similarity of a certain number of nearest neighbors. Its formula is

$$P_{u,i}=\bar{U}+\frac{\sum_{v\in N}\mathrm{sim}(u,v)\times(R_{v,i}-\bar{V})}{\sum_{v\in N}|\mathrm{sim}(u,v)|}. \quad (3)$$

$P_{u,i}$ is the prediction score of user u for the target item i.

After predicting the score, the prediction score is sorted, and the N items with the highest forecast score is selected as the recommended list to be recommended to the target user.

1.2 Challenges of User-Based Collaborative Filtering Algorithms

Collaborative Filtering (CF) is one of the most frequently used RS techniques. In CF, items are recommended to the target user through an analysis of most similar users' (neighbor users) ratings on those items. Though it is currently the most successful recommendation technique, it has weakness in dealing with the following challenges.

(a) There are a large number of users and items in large-scale e-commerce websites. The traditional collaborative filtering-based recommendation searches the entire neighbors of the target users or target items in the whole space, which is very time-consuming and poor scalability, and greatly affects the efficiency of the system;

(b) With the rapid development of e-commerce, the number of users and types of items also increase sharply, making the dimension of user-item ratings matrix increasing, most users have no or very few common rating items, which results in extremely sparse scoring matrices and a sharp drop in the recommended quality based on the traditional similarity if only considering user ratings;

(c) Traditional algorithms treat each user as an independent individual, ignoring the social nature of the user. If a user like an item, this item is more likely to be liked by the user they trust.

These problems are the challenges that the traditional collaborative filtering recommendation algorithm faces, which affect the recommendation performance and recommendation quality of the recommendation system. Many scholars have studied these problems.

For the sparsity problem of user-item ratings matrix, many researchers apply the idea of clustering to the collaborative filtering algorithm to alleviate the data sparsity problem. Yu and Li (2010) and Shinde and Kulkarni (2012) reduced the rating data by clustering technology, in order to reduce the search space of target user's nearest neighbor and improve the prediction accuracy. J. Kelleher and D. Bridg (Kelleher & Bridge, 2003) proposed a method which uses the k-means clustering algorithm to group similar users that can help to make reliable and scalable recommendations. A highly scalable algorithm which used a specific variant of the *k*-means clustering algorithm in the CF approach was proposed in Ref (Rashid & Lam, 2006).

In addition to user similarity, other factors also play important roles in providing high quality recommendations. Trust statements have recently been identified as effective means to utilize the social network and improve the recommendation quality. Recent works have proved that incorporating social factors or trust information in recommender systems has several benefits in terms of improving the quality of recommendations (Kitisin & Neuman, 2006). In Ref (Abdul-Rahman & Hailes, 2000), it has been shown that a user generally develops his social connections with someone who has similar tastes. Moreover, it has been shown that using trust statements can effectively improve the accuracy of recommender systems in comparison with pure CF algorithms (Golbeck, 2006).Various techniques have been proposed to employ trust information into the CF approaches (Jamali & Ester, 2010). Based on the social trust network, Avesani et al. (Navgaran & Moradi, 2013) used a certain length of path value to calculate the trust value between the target user and other users. In Ref (Massa & Avesani, 2007) the authors reported similar results and showed that adding social network data to traditional CF improves recommendation results. In Ref (Bedi & Sharma, 2012) a trust-aware method known as TARS is proposed to produce valuable recommendations by incorporating a notion of dynamic trust between users. The authors of Ref (Jamali & Ester, 2009) proposed the so-called Trust- Walker, which performs a random walk in online social networks to query a user's direct and indirect friends' ratings for the target item.

Through researching and analyzing related literatures, we propose a collaborative filtering recommendation algorithm based on user preference clustering and improved similarity weight introducing trust weight.

2. PROPOSED METHOD

In this paper, an improved collaborative filtering method by using a user preference clustering algorithm is proposed. In this section, details of the proposed method are described. The proposed method consists of four phases which are shown in Figure 1. In the first phase, an user-preference matrix is constructed to reduce the data sparsity. Based on the matrix, we know the similarity among users based on user's preference. Then, in the second phase, a *k*-means clustering algorithm is applied to group the users with similar preference into several clusters. This phase consists of four steps which includes: Finding initial cluster centers, calculating the distance between the center and the other user, Modifying cluster center sand Merging clusters. In the third phase, trust information integrating with user's preference is also considered in the similarity computations. Finally, in the fourth phase, for each unseen item, a rate is predicted and the top-N interested items are recommended to the active user. This phase includes two steps called Rate prediction, and top-N recommendations. The aim of the first and the second phases is to construct a model and the third phase attempts to predict the unknown rates for the active user and suggest him/her a list of interesting items. In this mode, the clustering algorithm is applied to the users and trust information of the users is also considered in the similarity computations. This mode is known as User Preference Clustering for Trust-Aware Recommender System.

2.1 Constructing User-Preference Matrix

According to the user's preference for different types of items, the user-preference matrix is constructed, so the data dimension is reduced. Then users are clustered by *k*-means clustering algorithm. Users with similar interests are divided into the same cluster, thus it can reduce the amount of computation and complexity effectively.

2.1.1 Item-Attributes Matrix

An item contains many categories of attributes, such as a movie that can be both a romance and a feature film. We represent the relationship between the same types of items as an $n \times g$ matrix called the item attribute matrix A_{ng}, which contains g types of n items, each of which is a Boolean, to represent "Item *i* belongs to type *j*", expressed as:

$$A(i,j)=\begin{cases}1, & \text{item } i \text{ belongs to type } k\\ 0, & \text{item } i \text{doesn't belongs to type } k\end{cases}, \tag{4}$$

where $A\ (i, j) = 1$ denotes that the item *i* belongs to type *j*, and $A\ (i, j) = 0$ means that the item *i* does not belong to type *j*. We can construct the item-attribute matrix based on the above formula.

Table 1
Item-Attributes Matrix

Item	$Attr_1$	…	$Attr_i$	…	$Attr_k$
$Item_1$	0	…	1	…	0
…	…	…	…	…	…
$Item_i$	1	…	0	…	1
…	…	…	…	…	…
$Item_m$	0	…	1	…	1

2.1.2 User-Preference Matrix

Definition 1. User Preference: Preferences are emotional tendencies that exist in people's minds with some subjective knowledge and preference perceptions. User preference indicates the user's preference for certain types of items, which is calculated as follows:

$$L(u,i)=\frac{\text{all Score}(u,i)}{\text{all Score}(u)}, \tag{5}$$

where all Score (*u, i*) denotes the sum score of user *u* for all the items with attribute *i*, and all Score (*u*) denotes the sum score of user *u* for all the items. According to the Formula (4), the users' preference for all project types can be expressed as a two-dimensional matrix, which is user-preference matrix.

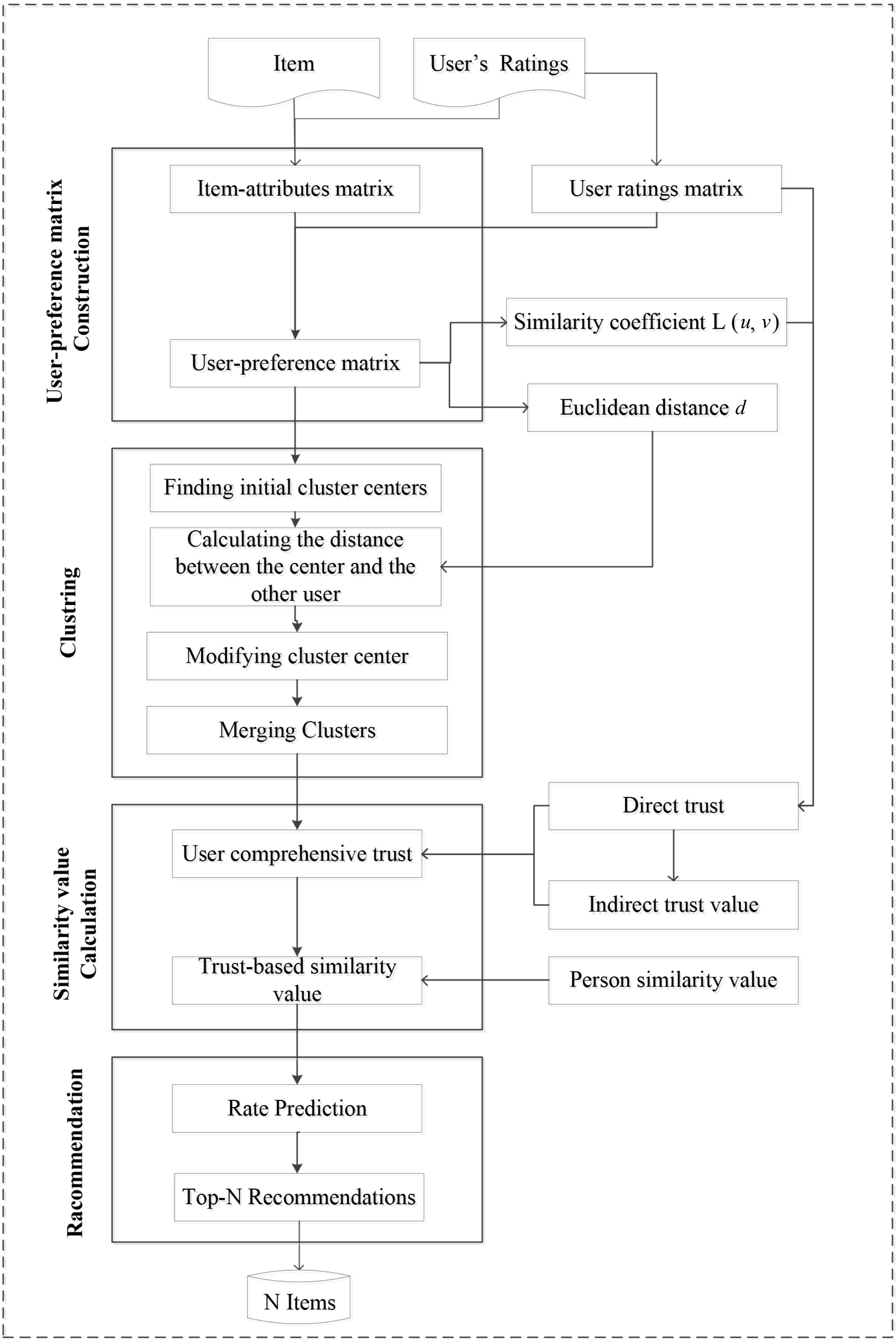

Figure 1
Overview of the Proposed Method

Table 2
User-Preference Matrix

User \ Attr	Attr$_1$	…	Attr$_i$	…	Attr$_k$
User$_1$	$L(1,1)$	…	$L(1,i)$	…	$L(1,k)$
…	…	…	…	…	…
User$_u$	$L(u,1)$	…	$L(u,i)$	…	$L(u,k)$
…	…	…	…	…	…
User$_m$	$L(m,1)$	…	$L(m,i)$	…	$L(m,k)$

2.1.3 Distance and Similarity Calculation Based on User's Preference

Based on the user-preference matrix, methods for calculating similarity values the Euclidean distance and between users are introduced.

We proposed the similarity coefficient $L(u, v)$ of user u and v based on the user-preference matrix is calculated as follows:

$$L(u,v)=\frac{\sum_{i=1}^{k}L_{u,i}L_{v,i}}{\sqrt{\sum_{i=1}^{k}{L_{u,i}}^2}\sqrt{\sum_{i=1}^{k}{L_{v,i}}^2}}, \tag{6}$$

where $L_{u,i}$ and $L_{v,i}$ denote user u and v's preference for an item with attribute i.

Assuming that there are k attributes of items in the proposed system, the k-dimensional vectors $\overrightarrow{x_u}$ and $\overrightarrow{x_v}$ denote the user u and v's preference values to k attributes. Euclidean distance between user u and v based on the user's preference is calculated as follows:

$$d_{u,v}=\sqrt{\sum_{k=1}^{m}\left|x_{u,k}-x_{v,k}\right|^2}, \tag{7}$$

where $x_{u,k}$, $x_{v,k}$ denote the user u and v the preference value of item with attribute k.

2.2 Clustering Based on User Preference

In this section user preference clustering method is proposed to group the users into several clusters. The proposed clustering method consists of three steps including (a) Finding initial cluster centers, (b) Modifying cluster center sand (c) Merging clusters.

The clustering results depend on the similarity measures of clustering objects. Euclidean distance based on user-preference matrix has been proposed. Users with similar preferences are divided into the same user cluster. The similarity measure of clustering adopts the Euclidean distance Formula (5). Assuming that the number of users in a cluster is N and the item has m attributions, the specific process of clustering a user is described as follows:

(a) Selecting k users $u1,u2,\ldots,uk$ from the users randomly as the initial cluster and putting the k user's preference values for m attributions $L(u_i, j)$,$i=\{1,2,\ldots,k\}$, $j=\{1,2,\ldots,m\}$as the initial cluster centers.

(b) Calculate the distance between the user's preference value of each remaining user v and the cluster center, which will be divided into the most similar cluster C_m.

(c) Readjust the cluster center of each cluster based on the user set in each cluster, and take the average of the user's preference values contained in C_m as the new cluster center.

(d) After the adjustment in step (3), if the adjusted new cluster center is the same as the last cluster center, or after the adjustment, if the error of the cluster center is less than a certain threshold, the clustering ends; Otherwise returning to step (2) to continue.

In order to prevent the endless loop happen when the termination condition of step (4) cannot be satisfied, a fixed maximum number of iterations can be set during clustering.

2.3 Trust-Based Similarity Value

The traditional recommendation process based on user-based collaborative filtering algorithm considers only the user-item rating matrix. It ignores the impact of user trust relationship and time factor, so the recommendation is not effective. This section proposes a collaborative filtering recommendation model that combines trust confidence. The basic idea is to integrate the trust relationship into the traditional similarity calculation method based on common score of users, and to further accurately measure the inter-user similarity, so as to determine the target user's nearest neighbor more accurately.

2.3.1 User Comprehensive Trust Based on Trust Relationship

Definition 2 Trust (TR): In the user rating system, if two users rate the same item to generate an interaction, this shows that there is a certain degree of trust between the two users. Calculated value means the trust between users.

Wang (2014) provides an effective method of calculating trust by allowing both recommended users and target users to participate in the project, and calculating the trust among users based on the number of successful interactions with the project score. In trust model building, use Equatuin (4) to represent the initial trust level.

$$\mathrm{In}iT(u,v)=\frac{\min\,(I_u\cap I_v,d)}{d}, \tag{8}$$

where $I_u \cap I_v$ denotes the number of times that user u and user v have interacted, and threshold d denotes the minimum number of times that the two trust each other completely. If the number of user interaction items does not exceed the threshold, the threshold d will play a role. If the number of common score items between users exceeds the threshold d, the initial trust level is 1. It is unrealistic to measure the degree of trust between users only by the number of user's rating interactions. Assuming that user u and user v have a large number of scoring interactions, but there are many differences between the scores. If you also use Formula (4) to denote the direct trust between users, there will be a great deviation to measure trust. Therefore, introducing the evaluation factor based on Formula (4), as shown in Formula (5).

$$\operatorname{correct}(u, v, c)=\begin{cases}1, & \left|R_{u,c}-R_{v,c}\right| \leq \varepsilon \\ 0, & \left|R_{u,c}-R_{v,c}\right|>\varepsilon\end{cases} . \quad (9)$$

If the difference of rating between user u and user v on item c is not greater than ε, it denotes that this interaction is successful and the corresponding number of successful interactions is accumulated as (count_s+1). Otherwise, the corresponding number of successful interactions stays constant. In the process of user interaction, the initial trust value is adjusted, and the direct trust value $T(u, v)$ is obtained as shown in Formula (6).

$$T(u, v)=\operatorname{Ini} T(u, v) \frac{\text{count}_s}{\text{count}}, \quad (10)$$

where count_s denotes the number of successful interactions, count denotes the total number of interactions.

The collaborative filtering algorithm based on trust relationship usually build trust model with simple user interaction times, and then combine with collaborative filtering algorithm to provide predictive recommendation to users. However, the above method based on interactive behavior weakens the influence of individual differences on interactive behavior, and does not consider the potential factor that affects the trust relationship - user preference. Therefore, this paper proposes a comprehensive trust value based on user preference method.

Definition 3: Trust relationship: A subjective cognition of the trustworthiness and validity of the recommended user by the system user through direct and indirect relationship.

Definition 4: Direct trust: Describing the user's direct trust according to the user's rating of the same item and user preference.

User preference: we have defined it in 3.1.2 Definition 1

According to the above definition of trust relationship, we need to obtain the trust value of two users. So we can provide the trust data for the recommendation algorithm. In the process of constructing the trust model, trust can be classified in direct trust and indirect trust according to the way we obtain trust. Therefore, this paper also divides trust in recommendation into direct trust and indirect trust. However, this paper proposed a method which is different from the traditional trust model. We incorporate the user's preference in the calculation of direct trust value to reflect the influence of the difference of user's preference for the user's similarity of ratings. So that the construction of trust model is more complete. We calculate indirect trust value based on direct trust value and the friend relationship through mutual interaction. We will describe the method in detail.

The trust model described in Formula (6), whether successful or unsuccessful, the user score interaction, has a weight of 1, without a finer measure of the quality of the interaction. Therefore, by incorporating the user's preference into the calculation of the direct trust, we use Formula (7) to represent the user's preference of user u for item c and its formula.

$$F(u, c)=\frac{\sum_{m \in U_c} L(u, v)}{\left|U_c\right|}, \quad (11)$$

where U_c denotes the set of users scoring item c, m denotes the user in U_c, $L(u, v)$ denotes the user preference similarity between user u and user v, as shown in Formula (5). Here we use the user preference similarity value to measure the user's preference. When user u and user U_c have higher similarity value, the score between users is more credible and evaluation weight should be greater. Therefore, user u has a higher degree of preference for item c. Conversely, the user preference is lower.

Definition 5 Indirect trust: Indirect trust refers to a trust relationship between two users through several direct trust relationships between other related users.

Among them, the user indirect trust relationship is formed by the direct trust relationship path connection. The indirect trust value IDT (u, v) is obtained as shown in Formula (12)

$$IDT\ (u, v)=\frac{\sum_{m \in S} DT(u, m) DT(m, v)}{\sum_{m \in S} DT(u, m)}, \quad (12)$$

where m denotes the direct trusted user of user u and S is the direct trusted user set of user u.

Definition 5 Comprehensive trust: It is a combination of direct trust and indirect trust, which is a comprehensive measure of the trust relationship between users. In comprehensive trust value, the weights of direct trust and indirect trust are coordinated by using dynamically generated weighting factors. The range of weights is [0,1], as shown in Formula (13).

$$\delta=\frac{DT(u, v)}{DT(u, v)+IDT(u, v)} . \quad (13)$$

Finally, the method of adjusting parameters is used to combine the direct trust with the indirect trust to calculate the comprehensive trust, as shown in Formula (14).

$$IT(u,v)=\delta DT(u,v)+(1-\delta) IDT(u,v) . \quad (14)$$

2.3.2 Trust-Based Similarity Value

Combined with user's comprehensive trust and rating similarity, an improved similarity calculation method is proposed, which makes the recommendation more accurate and reliable. We use the commonly used weighted mixing method to calculate the final weight, as follows:

$$C(u,v)=,\begin{cases}0 & IT(u, v) \leq 0 \\ \mu IT(u, v)+(1-\mu) \operatorname{sim}(u, v) & IT(u, v)>0\end{cases}, \quad (15)$$

where IT (u, v) denotes user's comprehensive trust in (14), and sim (u, v) denotes user's rating similarity. We determine the final parameter by constantly adjusting the parameter μ, so that the algorithm can achieve the best accuracy. Then we will find nearest neighbors of the trust-based similarity value.

2.4 Recommendation

In this section a set of interesting items is recommended to the active users based on the clusters and nearest neighbors found in the previous step. To this end, in the rate prediction step, the rate of item i for user u is generally predicted using the following Equation (13):

$$P_{u,i} = \overline{U} + \frac{\sum_{v \in S} C(u,v)(R_{v,i} - \overline{V})}{\sum_{v \in S} C(u,v)}, \tag{16}$$

where $C(u,v)$ denotes the trust-based similarity value in Formula (15), the other denotes the same as Formula (3).

Finally, in the top-N recommendations step, the algorithm predicts the rates of the unseen items for the active user and then selects those of the top-N items to be recommended.

2.5 The Pseudo Code of the Proposed Method

The pseudo code of the proposed method is described in Table 3.

Table 3
The Pseudo Code of the Proposed Method

Algorithm: A trust-based collaborative filtering algorithm using a user preference clustering
Define: User-item matrix *R*, Item set *I*, Target user *u*, Neighbor collection *N*, Predictive score *P*, User set *U*, User Similarity *C*, Trust *IT*
Input: *R , I, K, U*
Output: top-N recommendation list
Algorithm:
1. Build User-preference matrix *Y* with the user preference vector =(,) use function (5)
2. Random select *K* users from *U*, Taking the *K* users of the user preference vector as the initial *K* cluster centers $CC=\{cc_1,cc_2,\ldots,cc_k\}$
3. Repeat
4. For each user $u_i \in U$
5. For each cluster center $cc_j \in CC$
6. Caculate $d_{u,cc}$ use founction（7）
7. End for
8. $d(i_i ,cc_m)$=max $\{d(u_i, cc_1), d(u_i, cc_2),\ldots, d(u_i, cc_k)\}$
9. Cluster $=c_m=c_m \cup u_i$
10. End for
11. For each cluster $c_i \in c$
12. Calculate the cluster c_i in all users of the average user preference and generate a new cluster center cc_j
13. End for
14. Until all cluster centers are the same as the cluster centers in the previous cycle (the members in the cluster no longer change)
15. For $i \leftarrow 1$, user number M do
16. For $j \leftarrow 1$, user number M-1 do
17. Caculate similarity *S* use Pearson algorithm
18. Trust *T* use Function (14)
19. Comprehensive similarity C use Function (15)
20. Select Top-*K* Neighbor collection *N*
21. Predict item set I score *P* use Function (16)
22. End for
23. End for
24. Recommend top-N of items as the recommendation list to the user *u*

In the proposed method, Line 1 is to construct user-preference matrix. Line 2-14 is to cluster the users with similar reference. Line 15-19 is to calculate user's similarity value, user's comprehensive trust value and user's trust-based similarity value. Line 20 selects *K* nearest neighbor sets in the target user's cluster. In line 21 and 24, the algorithm predicts the rates of the unseen items for the active user and then selects those of the top-N items to be recommended.

3. EXPERIMENTS

In order to verify the proposed method in this paper, several experiments are performed in this section. Moreover, the proposed method with user-based has been compared with the traditional user-based collaborative filtering, trust-aware clustering collaborative filtering (Wang, 2014).The detailed descriptions of the employed datasets, evaluation measures and obtained results are mentioned in the corresponding sections. The experiments have been run on a machine with a 2.2 GHz CPU and 8GB of RAM. Moreover, all of the methods were implemented using python programming language.

3.1 Datasets

The MovieLens dataset was collected by the GroupLens research group (Resnick & Iacovou, 1994) and consists of 100,000 ratings from 943 users on 1682 different movies. The ratings are integer numbers in the range of 1 (bad) to 5 (excellent) scales. Each user in this dataset has rated at least 20 movies.

3.2 Evaluation Measures

In this paper to evaluate the recommendation methods, each dataset is divided into a training set and a test set (Lü & Medo, 2012). The training set is treated as known information and then leave-one-out method (Massa & Avesani, 2007), is applied on the test set for evaluating performance of the recommendation algorithms. There are several accuracy based metrics in the literature including mean absolute error (MAE), root mean square error(RMSE), precision, recall and F1 measures (Ibid.), of which Mean Absolute Error (MAE) is one of the widely used in recommended systems. It measures the accuracy of the algorithm by means of the average difference between the predicted score and the actual score. Therefore, in this paper, MAE is used to compare the accuracy of the proposed method with the other methods.

The MAE is used to measure the closeness of predicted ratings for the true ratings. To compute the absolute error value, the predicted rate is compared with the real ones. This procedure is repeated for all of the taken-out ratings and then an average of all the values is considered as the MAE value as follows:

$$\mathrm{MAE} = \frac{\sum_{i=1}^{N} |r_i - p_i|}{N},$$

where r_i and p_i are actual and predicated rates of an item i, respectively, and N denotes the total number of rates that are predicted by a recommender method.

3.3 Results and Analysis

In this section, a number of experiments were performed to evaluate the usefulness of the proposed methods.

Experiment 1: To verify the effectiveness of the clustering method based on the k-means and user preference. The method divides users with similar preference into the same cluster. Then when we find nearest neighbors, we can find them just based on the result of this clustering. Because multiple iterations are needed to calculate the similarity between data objects and clustering centers, if the number of specified clusters is too large, the computational complexity of the system will increase. However, if the number of specified clusters is too small, we need to query for a large number of other users in each cluster. So it is difficult to set a proper value k for the clustering method, which may influence the final results of recommendation The experiment cluster the user set based on the number 30, 40, 50, the nearest neighbor is set to 30. In traditional collaborative filtering algorithm, the efficiency to query 30 nearest neighbors is as Figure 2 shows.

The results of Experiment 1 show that when the number of nearest neighbors is thirty, as the percentage of search user in the entire user space gradually increases, the percentage of neighbors gradually increases. The growth rate is getting lower and lower, tending to be stable. This happens because the 30 nearest neighbors of

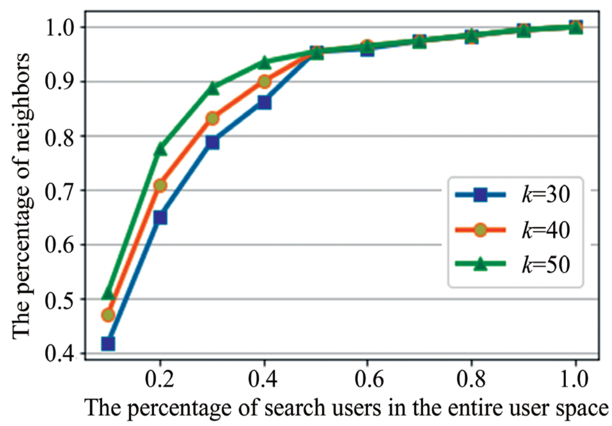

Figure 2
The Comparison of Nearest Neighbor Query Efficiency

the target user are almost divided into the nearest cluster of the target user. Among the user's clusters that are most similar to the target user, there are very few target user neighbors in other user clusters. When the size of the clusters number is 30, the user can find 80% nearest neighbors by searching only 33% of the entire user set, and the user can find 90% nearest neighbors by searching only 48% of the entire user set. When the size of the clusters number is 40 the user can find 80% nearest neighbors by searching only 30% of the entire user set, and the user can find 90% nearest neighbors by searching only 40% of the entire user set. When the size of the clusters number is thirty, the user can find 80% nearest neighbors by searching only 23% of the entire user set, and the user can find 90% nearest neighbors by searching only 38% of the entire user set. Experimental results show that the clustering algorithm can help the target user find as much neighbors as it can by searching less entire user space, which can improve the algorithm efficiency.

Experiment 2: To determine the minimum parameter d, the number of interactions between two users fully trusting each other in the initial direct trust calculation (8), we discuss the effect on MAE when we choose different parameter d=40,50,60. In the MovieLens dataset, the experimental results are as Figure 3 shows.

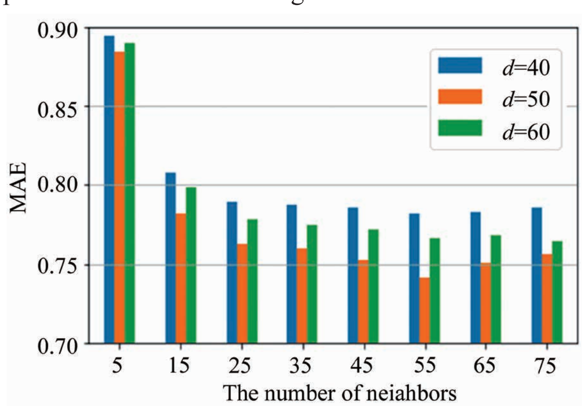

Figure 3
The Effect of Parameter d on the MAE

As Figure 3 shows, as the number of the nearest neighbor changing from 5 to 75, the MAE first decreases and then rises. When d = 50, the MAE is the smallest, which means that in the MovieLens dataset. When a user completely trust each other for a minimum of 50 interactions, the proposed algorithm achieves the best accuracy.

Experiment 3: In order to test the effect of the evaluation factor ε on the recommended accuracy in the calculation Formula (15), we set experiment in the Movie Lens data with ε = 1, 2 and 3, the result is shown in Figure 4.

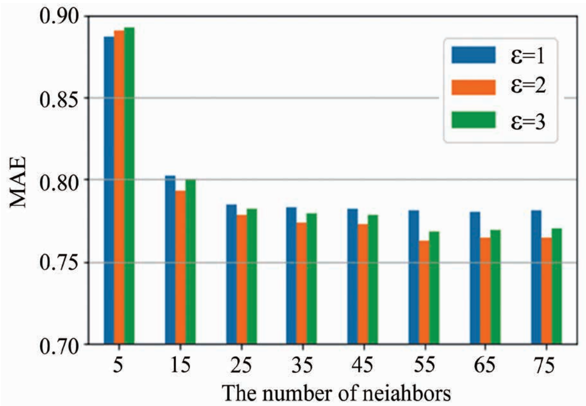

Figure 4
The Effect of Parameter ε on the MAE

It can be seen from Figure 4, when ε= 2, the MAE is the lowest among the same nearest neighbors. This indicates that when the rating of the two users interacting items is 2, the proposed algorithm achieves the best accuracy.

Experiment 4: In order to determine parameter μ in Formula (15), we compare the MAE by setting the parameters from 0 to 1. In the MovieLens dataset, the experimental result is as Figure 5 shows.

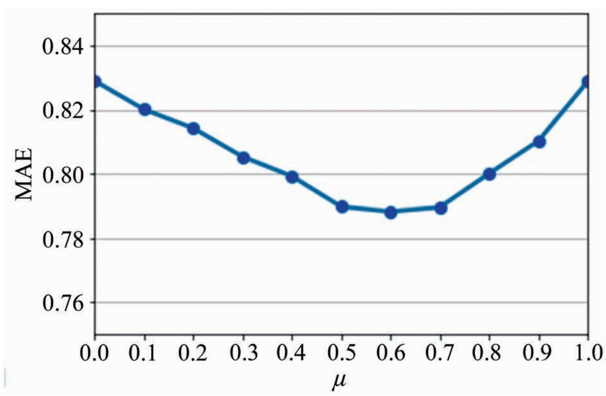

Figure 5
The Effect of Parameter on the MAE

We can know from Figure 5 that MAE first decrease then increases as the parameter μ changing from 0 to 1. When μ=0.6, MAE is the lowest, which means the proposed algorithm achieves the best accuracy.

Experiment 5: In order to compare the accuracy of different methods, we design an experiment in the MovieLens dataset. The two proposed methods are evaluated and compared with the methods including the traditional user-based collaborative filtering algorithm (UBCF), the traditional collaborative filtering algorithm based on trust relationship (TRCF) in Literature (Matrix et al., 2004), the proposed improved trust relationship (ITRCF) method, the proposed and the proposed trust-based collaborative filtering algorithm using a user preference clustering(K-means ITRCF). The result is shown in Figure 6.

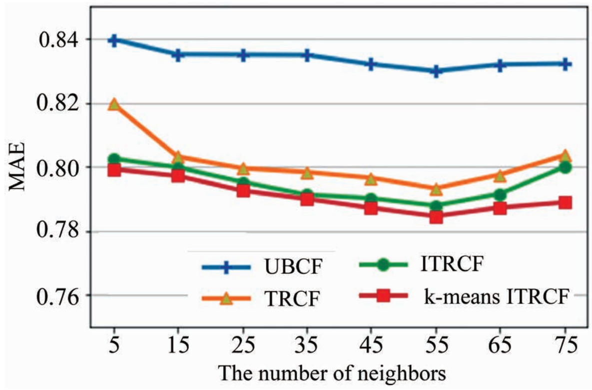

Figure 6
MAE Comparison of Different Collaborative Filtering Algorithms

The results in Figure 6 show that the ITRCF and K-means ITRCF method obtained better results on the MAE measures than the other traditional methods.

CONCLUSION

The collaborative filtering approach is a powerful technology for users to find their interesting information. Trust is a concept that has recently attracted much attention and has been considered in online recommendation systems. In this paper, a trust-based collaborative filtering method by using user preference clustering algorithm is proposed. The proposed method consists of four phases. In the first phase, an user-preference matrix is constructed to reduce the data sparsity. Based on the matrix, we know the similarity among users based on user's preference. Then, in the second phase, a k-means clustering algorithm is applied to group users with similar preference into several clusters. In the third phase, trust information integrating with user's preference is also considered in the similarity computations. Finally, in the fourth phase, for each unseen item, a rate is predicted and the top-N interested items are recommended to the active user.

The experimental results show that compared with the traditional user-based collaborative filtering algorithm and the traditional collaborative filtering algorithm based on trust relationship, the proposed improved algorithm achieves the best prediction accuracy (minimum MAE) on the test data set. Therefore, this model can effectively improve the accuracy of the score prediction and the quality of collaborative filtering recommendation.

The proposed clustering algorithm has used the user preference to reduce the sparsity, and introducing the user preference to the trust-based similarity value, so that the similarity calculation more accurate. Meanwhile, we with user preference clustering, this model can not only improve the efficiency but also the accuracy.

For the future research we can consider different factors to make more effective methods. For this purpose, incorporating distrust statements in the clustering algorithm may lead to improvement of the recommendation results. On the other hand, using fuzzy concepts such as fuzzy c-means is another way to improve clustering algorithms and can help to make better recommendations.

REFERENCES

Abdul-Rahman, A., & Hailes, S. (2000). *Supporting trust in virtual communities.* In 3th Hawaii International Conference on System Sciences, Hawaii, USA.

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering, 17*(6), 734-749.

Bedi, P., & Sharma, R. (2012). Trust based recommender system using ant colony for trust computation. *Expert Syst. Appl., 39,* 1183-1190.

Birtolo, C., & Ronca, D. (2013). Advances in clustering collaborative filtering by means of fuzzy c-means and trust. *Expert Syst. Appl., 40,* 6997-7009.

Bojnordi, E., & Moradi, P. (2013). A novel collaborative filtering model based on combination of correlation method with matrix completion technique. *Iran. J. Sci. Technol. Trans. Electr. Eng., 37,* 93-100.

Breese, J., Heckerman, D., & Kadie, C. (1998). *Empirical analysis of predictive algorithms for collaborative filtering.* Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence. Madison, USA: [s. n.].

Breese, J. S., Heckerman, D., & Kadie, C. (1998). *Empirical analysis of predictive algorithms for collaborative filtering* (pp.43-52). In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc., Madison, Wisconsin.

Castro-Schez, J., & Miguel, R. (2011). A highly adaptive recommender system based on fuzzy logic for B2C ecommerce portals. *Expert Systems with Applications, 38*(3), 2441-2454.

DuBois, T., Golbeck, J., Kleint, J., & Srinivasan, A. (2009). *Improving recommendation accuracy by clustering social networks with trust.* In ACM RecSys Workshop Recommender Systems and the Social Web.

Golbeck, J. (2006). *Generating predictive movie recommendations from trust in social networks*. In 4th International Conference on Trust Management Pisa, Italy.

Golbeck, J., & Hendler, J. (2006). *FilmTrust: Movie recommendations using trust in web-based social networks* (pp.282-286). In IEEE Consumer Communications and Networking Conference.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems, ACM Trans. *Inform. Syst., 22,* 5-53.

Jamali, M., & Ester, M. (2010). *A matrix factorization technique with trust propagation for recommendation in social networks*. In ACM Conference on Recommender Systems.

Jamali, M., & M. Ester, M. (2009). *Using a trust network to improve top-N recommendation* (pp.181-188). In Proceedings of the Third ACM Conference on Recommender Systems, Pub- lishing, New York, New York, USA.

Javari, A., Gharibshah, J., & Jalili, M. (2014). Recommender systems based on collaborative filtering and resource allocation. *Soc. Netw. Anal. Min., 4,* 1-11.

Javari, A., & Jalili, M. (2014). Cluster-based collaborative filtering for sign prediction in social networks with positive and negative links. *ACM Trans. Intell. Syst.Technol. (TIST) 5,* 24.

Kelleher, J., & Bridge, D. (2003). *Rectree centroid: An accurate, scalable collaborative recommender* (pp.89-94). In Proceedings of the Fourteenth Irish Conference on Artificial Intelligence and Cognitive Science, Citeseer.

Kitisin, S., & Neuman, C. (2006). Reputation-based trust-aware recommender system (pp.1-7). In Securecomm and Workshops.

Liu, B., & Yuan, Z. (2010). *Incorporating social networks and user opinions for collaborative recommendation: Local trust network based method* (pp.53-56). In Proceedings of the Workshop on Context-aware Movie Recommendation, ACM, New York, USA.

Lü, L., Medo, M., Yeung, C. H., Zhang, Y.-C., Zhang, Z.-K., & Zhou, T. (2012). Recommender systems. *Phys. Rep., 519,* 1-49.

Massa, P., & Avesani, P. (2004). *Trust-aware collaborative filtering for recommender systems.* In Federated Int. Conf on the Move to Meaningful Internet.

Massa, P., & Bhattacharjee, B. (2004). *Using trust in recommender systems: An experimental analysis* (pp.221-235). In Proceedings of 2nd International Conference on Trust Managment, Oxford, England.

Massa, P., & Avesani, P. (2005). *Controversial users demand local trust metrics: An experimental study on epinions. com community*. In American Association for Artificial Intelligence (AAAI) Conference, San Francisco, USA.

Massa, P., & Avesani, P. (2007). *Trust-aware recommender systems.* In 2007 ACM Conference on Recommender Systems, Minneapolis, Minnesota, USA.

Navgaran, D. Z., Moradi, P., & Akhlaghian, F. (2013). *Evolutionary based matrix factorization method for collaborative filtering systems* (pp.1-5). In 2013 21st Iranian Conference on Electrical Engineering, ICEE, IEEE.

Park, S. T., & Pennock, D. M. (2007). *Applying collaborative filtering techniques to movie search for better ranking and browsing* (pp.550-559). Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press.

Ramezani, M., Moradi, P., & Akhlaghian, F. (2014). A pattern mining approach to enhance the accuracy of collaborative filtering in sparse data domains. *Physica A, 408,* 72-84.

Rashid, A. M., Lam, S. K., Karypis, G., & Riedl, J. (2006). *Clustknn: A highly scalable hybrid model-& memory-based cf algorithm.* In Proceedings of WebKDD, Citeseer, Philadelphia, PA.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). *GroupLens: An open architecture for collaborative filtering of netnews* (pp.175-186). In Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work, ACM, Chapel Hill, North Carolina, USA.

Sarda, K., Gupta, P., Mukherjee, D., Padhy, S., & Saran, H. (2008). *A distributed trust-based recommendation system on social network*. In IEEE 10th Second Workshop Hot Topics in Web Systems and Technologies.

Shang, M.-S., Zhang, Z.-K., Zhou, T., & Zhang, Y.-C. (2010). Collaborative filtering with diffusion-based similarity on tripartite graphs. *Physica A, 389,* 1259-1264.

Sinde, S. K., & Kulkarni, U. (2012). Hybrid personalized recommender system using centering-bunching based clustering algorithm. *Expert Systems with Applications, 39*(1), 1381-1387.

Tsai, C. F., & Huang, C. H. (2012). Cluster, ensembles in collaborative filtering recommendation. *Appl. Soft Comput, 12,* 1417-1425.

Walter, F., Battiston, S., & Schweitzer, F. (2007). A model of a trust-based recommendation system on a social network, Auton. *Agents Multi-Agent Syst., 16,* 1573-7454.

Wang, J. H. (2014). *Collaborative filtering recommendation lgorithm combining trust mechanism with user preferences.* Chongqing University.

Xu, H. L., Wu, X., Li, X. D., & Yan, B. P. (2009). Comparison study of internet recommendation system. *Journal of Software,* (20), 350-362.

Yu, X., & Li, M. Q. (2010). Effective hybrid collaborative filtering model based on PCA-SOM. *System Engineering-Theory & Practice, 30*(10), 1850-1854.

Zhang, J., Peng, Q., Sun, S., & Liu, C. (2014). Collaborative filtering recommendation algorithm based on user preference derived from item domain features. *Physica A, 396,* 66-76.

