

Collaborative Filtering Recommendation Based on Trust Model with Fused Similar Factor

Li Ye¹, Chunming Wu^{1,*}, Min Li¹

¹College of Computer & Information Science, Southwest University, 400715, China

Abstract. Recommended system is beneficial to e-commerce sites, which provides customers with product information and recommendations; the recommendation system is currently widely used in many fields. In an era of information explosion, the key challenges of the recommender system is to obtain valid information from the tremendous amount of information and produce high quality recommendations. However, when facing the large amount of information, the traditional collaborative filtering algorithm usually obtains a high degree of sparseness, which ultimately lead to low accuracy recommendations. To tackle this issue, we propose a novel algorithm named Collaborative Filtering Recommendation Based on Trust Model with Fused Similar Factor, which is based on the trust model and is combined with the user similarity. The novel algorithm takes into account the degree of interest overlap between the two users and results in a superior performance to the recommendation based on Trust Model in criteria of Precision, Recall, Diversity and Coverage. Additionally, the proposed model can effectively improve the efficiency of collaborative filtering algorithm and achieve high performance.

1 Introduction

The tremendous growth in information and number of global Internet users pose many key issues for recommender system to solve. Take the following for example: extracting useful information from all the massive data, producing high quality recommendations for the user.

Collaborative Filtering (CF) is currently the most widely used, personalized recommendation technology. The traditional collaborative filtering algorithm attempts to exploit the similarity between users who have the same interests and use the similarity to predict users' interests. Thus, it works by building a dataset of users' similarity that is based on the users' past actions and interests, then uses the user similarity to determine a neighbor set. A new user who is matched against the neighbor set discovers neighbors, which is the algorithm used to make the recommendation to the user. However, this algorithm has its own problems -- cold start, sparse data and low scalability [6]. At the same time, the accuracy of similarity between users is the key factor affecting the recommended performance. To improve the accuracy of the similarity between users, scholars have done a number of corresponding studies. Wu et al. [1] proposed a ratio-based approach to calculate the similarity between users. Chen et al. [2] proposed an improved algorithm based on the optimization of user similarity was proposed by adding the equilibrium factor to the traditional Cosine Similarity algorithm. Bobadilla et al. [3] used the traditional similarity to calculate the degree of preference of two users for one item, and the

greater the degree of preference implies a greater coherence, which is equivalent to an optimization similarity strategy. Huang et al. [4] studied the similarity between users by structural similarity in complex networks. BK Patra et al. [5] a similarity measure based on neighborhood CF was proposed. It used the rating made by neighboring users. Guo et al. [6-8] improved the similarity degree by adding the machine learning model on the basis of collaborative filtering algorithm, such as trust model, clustering model, feature model and association rule model. Although the methods mentioned above can improve the user similarity measure to a certain extent, they do not take into account the social trust in the similarity of users. To target this problem, this paper proposes a novel method: collaborative filtering recommendation based on Trust Model with Fused Similar Factor (TMFSF), which achieves higher precision and higher performance in making recommendations to users.

2 The user-based collaborative filtering

Collaborative filtering algorithm is divided into user-based collaborative filtering and item-based collaborative filtering. Among them, user-based collaborative filtering algorithm mainly includes the following three steps:

First, establish the user model, which namely is the users - items matrix. Suppose there are m users and n items, you can get a scale $m \times n$ matrix. As shown in Table.1, Where R_{ij} represents the ratings of user U_i for item I_j , and R_{ij} can be empty.

* Corresponding author: springsun@swu.edu.cn

Table 1. The users - items matrix.

	I_1	I_2	...	I_j	...	I_n
U_1		2				
U_2		4			3	
...	3					5
U_i		3		R_{ij}		
...				5		2
U_m		5				

Secondly, find the nearest neighbor of the target user. According to the user's rating on the project, find the similarity $Sim(U_i, U_j)$ between the target user and other users. Then, descend neighbors according to the similarity value, and get the nearest neighbor set by a given similarity threshold or "nearest" neighbors. The calculation method of user similarity mainly includes Cosine Similarity, Pearson Similarity and Modified Cosine Similarity. Among them, the Modified Cosine Similarity method is described in the formula (1):

$$Sim(U_i, U_j) = \frac{\sum_{c \in I_{i,j}} (R_{i,c} - \bar{R}_i)(R_{j,c} - \bar{R}_j)}{\sqrt{\sum_{c \in I_{i,j}} (R_{i,c} - \bar{R}_i)^2} \sqrt{\sum_{c \in I_{i,j}} (R_{j,c} - \bar{R}_j)^2}} \quad (1)$$

Where $Sim(U_i, U_j)$ is the similarity between user U_i and user U_j . $R_{i,c}$ and $R_{j,c}$ are the ratings of user U_i on item C . \bar{R}_i and \bar{R}_j are the average ratings of user U_i and user U_j respectively. $I_{i,j}$ indicates the item that user U_i and user U_j 's collectively ratings.

Thirdly, predict the ratings and generate a recommended list. Evaluate the target user's rating on the unrated item according to the "nearest" neighbor's ratings, generating a item recommendation set. Assuming that the number of "nearest" neighbors of the target user U_i is n . The predictive method of ratings is shown in formula (2).

$$P_{i,c} = \bar{R}_i + \frac{\sum_{m=1}^n Sim(U_i, U_j)(R_{j,c} - \bar{R}_j)}{\sum_{m=1}^n Sim(U_i, U_j)} \quad (2)$$

3 Trust Model with Fused Similar Factor

When facing a decision, we are more likely to accept viewpoints from credible sources. Hence, it is a valuable opportunity to improve recommendation quality by sufficiently and effectively utilizing trust information. The introduction of a trust model based on the trust relationship can adjust the weight of the similarity in the recommendation process [11], which in turn can eliminate the effectiveness caused by the sparseness original matrix, and improve the recommended quality to a certain extent.

In the traditional user-based collaborative filtering, the matrix of ratings is usually sparse and measurement

of users' similarity always has low precision, the recommended quality is generally not satisfactory. However, the trust model does not have this limitation because it eliminates the precision error in measuring users' similarity

Basing on the above considerations, this section gives the "trust model" and "similar factor" definition firstly. And then, introduce the similar factor combining with the Modified Cosine Similarity. Finally, introduce the similarity of fusing similar factor combining with the trust model, which is used as the comprehensive weight to replace the similarity in traditional algorithm. According to the ideas in the paper [9], similarity expressed the degree of coincidence of interest. The similar factor of introduced by this paper is used to improve the accuracy of the similarity, and then combined with the trust model. The harmonic weight is calculated replace the similarity, so as to improve the recommendation quality.

Trust Model with Fused Similar Factor specific process shown in Fig.1.

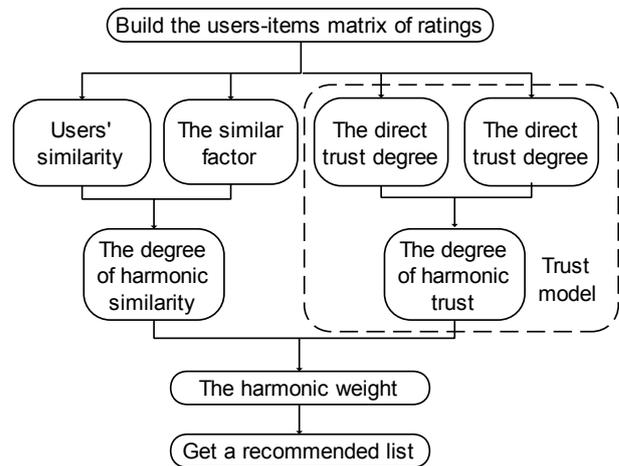


Fig. 1. Trust Model with Fused Similar Factor.

3.1 The Trust Model

The trust model is classified into direct trust degree and overall adjustment of trust degree [10]. The direct trust degree is the difference of two individual users' interests that is calculating directly by the users-items matrix. The overall adjustment of trust degree represents that user is trusted degree in the overall system. It is used to revise the direct trust. This paper combines of the direct trust degree and the overall adjustment of trust degree to get the degree of harmonic trust.

3.1.1 The Direct Trust Degree

In order to quantify the direct trust degree, this paper draws on the idea of the paper [10], using the formula (3) to calculate the direct trust degree $Trust1(U_i, U_j)$.

Definition 1: the direct trust degree [10].

$$Trust1(U_i, U_j) = \frac{Correct(U_i, U_j)}{C_{ij}} \quad (3)$$

where $Correct(U_i, U_j)$ is the correct number recommended by user U_j for user U_i , within the range of the number of common items rated by user U_i and user U_j . $C_{ij}=|C_i \cap C_j|$ represents the number of user U_i and user U_j common items of ratings.

The so-called "correct recommendation" means that the user U_j speculates that the speculative ratings of the user U_i generates a difference from the actual ratings of the user U_i . When the difference is less than the threshold e , it is considered to be the correct recommendation, see the formula (4).

$$\begin{cases} TR_{i,s} = R_i + (R_j - R'_j) * Sim'(U_i, U_j) \\ |TR_{i,s} - P_i| < e \end{cases} \quad (4)$$

where $TR_{i,s}$ denotes that user U_j estimates the speculative ratings of user U_i on item s . R_i represents the average ratings of user U_i . R_j represents the speculative rating of the item s by the user U_j . R'_j represents the actual rating of the item s by the user U_j . $Sim'(U_i, U_j)$ represents the harmonic similarity of user U_i and user U_j . P_i represents the actual rating of the item s by the user U_i . When e is a fixed value, when it is closer to 0, the correct recommendation is more accurate.

3.1.2 The Overall Adjustment of Trust Degree

In the recommendation system, there is a phenomenon that some users maybe become a lot of people "recent" neighbor, and some users maybe become very few people "recent" neighbors [10, 11]. It can be argued that the credibility of active users in the system is higher than that of inactive users. In calculating the trust degree of the target user to the neighbor user, in addition to considering the direct trust degree, it is also necessary to consider the performance of the neighbor user in all users [10].

Definition 2: The overall adjustment of trust degree. Set f_i be the total number of time s user U_i have evaluated in the system. q_i is the number of times a user has made a recommendation for other users. That is, the number of times the user become the "nearest" neighbor of other users.

$$Trust2(U_i) = \frac{2}{\frac{1}{1-1/\ln f_i} + \frac{1}{1-1/\ln(q_i+3)}} \quad (5)$$

3.1.3 The Degree of Harmonic Trust

The direct trust degree and the overall adjustment of trust degree combined to get the degree of harmonic trust, see the formula (6).

$$T(U_i, U_j) = \frac{2 * Trust1(U_i, U_j) * Trust2(U_i)}{Trust1(U_i, U_j) + Trust2(U_i)} \quad (6)$$

Where $T(U_i, U_j)$ represents the degree of harmonic trust between user U_i and user U_j , $Trust1(U_i, U_j)$ represents the direct trust degree of user U_j to user U_i , $Trust2(U_i)$ represents the overall adjustment of trust degree of user U_i to user U_j .

3.2 The Similar Factor

User's similarity indicates the degree of similarity between the user's preferences for the item. The accuracy of the similarity will directly affect the user's forecast ratings. Since the matrix of ratings is usually very sparse, there is insufficient information to deal with, which can lead to a significant reduction in the accuracy of similarity. If only calculating the similarity according to formula (1) will result in a large error [8]. Assume that two users rate items that are of interest to a group of items. It is obvious that if the number of common items rated by two people is more, the closer the interest preferences of the two users are. That is, the greater the similarity of users. The algorithm that introduces the trust model does not deal with the similarity between two users. Therefore, the similarity is introduced to preprocess the similarity.

Definition 3 : User ' s similar factors. Set $C_A=\{c_i|i \in [1, N_1]\}$ be the list of items evaluated by user U_A . Set $C_B=\{c_j|j \in [1, N_2]\}$ be the list of items evaluated by user U_B . $C_{AB}=|C_A \cap C_B|$ represents the number of user U_A and user U_B common items of ratings. S_{AB} represents the average of the user U_A and user U_B common items of ratings. Define the user similar factor, see formula (7).

$$\gamma_{ab} = \frac{C_{AB} + \alpha}{S_{AB} + \alpha} \quad (7)$$

Where α is a parameter of similar factor. Its value is to avoid the above formula in the denominator is 0 lead to the meaning of the formula, and it is also possible to prevent the value of γ_{ab} from 0 to cause the similarity to be 0. γ_{ab} represents a similarity factor between two users.

Based on the similar factor γ_{ab} , this article revises the user similarity, Based on the similar factor R , this paper revises the user similarity, named as the degree of harmonic similarity $Sim'(U_i, U_j)$, see formula 8.

$$Sim'(U_i, U_j) = \frac{2 * Sim(U_i, U_j) * \gamma_{ij}}{Sim(U_i, U_j) + \gamma_{ij}} \quad (8)$$

The degree of harmonic similarity $Sim'(U_i, U_j)$ replaces the similarity as the measure of the final users' similarity.

3.3 The Harmonic Weight

In the premise of revising the similarity of users, the introduction of trust model, the combination of the two can take full account of the user in the purchase of goods when the actual situation.

In this paper, combine the degree of harmonic similarity with the degree of harmonic trust, and get the harmonic weight instead of $Sim(U_i, U_j)$ in the formula (2), see the formula (9).

$$Weight(U_i, U_j) = \frac{2 * Sim'(U_i, U_j) * T(U_i, U_j)}{Sim'(U_i, U_j) + T(U_i, U_j)} \quad (9)$$

Where $Weight(U_i, U_j)$ represents harmonic weight. $Sim'(U_i, U_j)$ represents the degree of harmonic similarity with the similar factor, represents the degree of harmonic trust.

3.4 Generate Recommendations

The traditional user-based collaborative filtering algorithm in the similarity replaced by the harmonic weight. The predicted ratings are obtained according to the formula (10), and the *TopN* recommendation list is generated according to the predicted ratings.

$$P_{i,c} = \bar{R}_i + \frac{\sum_{m=1}^n Weight(U_i, U_j)(R_{j,c} - \bar{R}_j)}{\sum_{m=1}^n Weight(U_i, U_j)} \quad (10)$$

4 Trust Model with Fused Similar Factor

By introducing the above ideas into the traditional user-based collaborative filtering algorithm, an improved weighted cooperative filtering algorithm based on trust model is proposed. The algorithm firstly improves the accuracy of similarity, and then combines with the trust model to get the harmonic weight.

4.1 Algorithm Description

The following gives the algorithmic description of the degree of harmonic trust, the degree of harmonic similarity and harmonic weights.

Algorithm 1 Implement the degree of harmonic similarity

Input: *Sim_matrix* (The matrix of similarity between users)

tot_matrix(The total matrix of ratings)
 α (The parameter of similar factor)

Output: *Sim'(U_i, U_j)* (The degree of harmonic similarity)

- (1) On the basis of the matrix of the user's ratings, the similarity *Sim(U_i, U_j)* of the target user *U_i* and the other user *U_j* ($j \in [1, n]$, $U_i \neq U_j$) is obtained.
- (2) The items of ratings for users *U_i* and *U_j* are obtained respectively, denoted as $itemI = \{k \mid item_k > 0, k \in Z, k \in itemNumber\}$ and $itemJ = \{n \mid item_n > 0, n \in Z, n \in itemNumber\}$.
- (3) Use *Count* to record the number of common items rated by user *U_i* and user *U_j*, and if not, then *Count* is zero.
- (4) Use *Average(Sum(itemI), Sum(itemJ))* to record the average of user *U_i* and user *U_j* common rating items.
- (5) Calculate the similar factor $\gamma(U_i, U_j)$ according to formula (3).
- (6) Calculate the degree of harmonic similarity *Sim'(U_i, U_j)* according to formula (4).

Algorithm 2 Implement the degree of harmonic trust and harmonic weight

Input: *tot_matrix* (The total matrix of ratings)
bnc(The number of "nearest" neighbors per user)
correct(The correct number of recommendations)

Output: *Weight(U_i, U_j)* (Harmonic weight)

- (1) Calculate the correct number of users *U_i* based on the matrix of the user's ratings. According to the formula (6) and its evaluation criteria, the correct evaluation number *Correct(U_i, U_j)* ($j=1, 2, \dots, i-1, i+1, \dots, n$) is obtained.
- (2) According to the formula (5) to obtain direct trust degree *Trust1*.
- (3) Calculate user *U_i* to be the number of nearest neighbors for all other users, as $bnc = becomeNeighborCount(U_i)$.
- (4) Get a collection of user *U_i* rated, as $itemI = \{k \mid item_k > 0, k \in Z, k \in itemNumber\}$. The total evaluation number is calculated according to the item rating of user *U_i*, as $ReNumber = Sum(itemI)$.
- (5) According to step (1), step (3) and formula (7) to get the user to the overall adjustment of trust degree *Trust2(bnc, ReNumber)*.
- (6) According to the formula (8) to get the degree of harmonic trust *Trust(Trust1, Trust2)*.
- (7) According to the formula (9) to get the harmonic weight *Weight(Sim'(U_i, U_j), Trust)*.

4.2 Metrics

In order to validate the algorithm, this paper uses the Precision[13], Recall[14], Diversity[16] and Coverage[17] to analyze the experimental process. And use MAE (Mean Absolute Error) [12] and RMSE (Root Mean Square Error) as the measure of similarity factor α .

The Precision is in relation to the predicted results, which refers to the percentage of the extracted correct information that accounts for all the recommended information. The Recall is in relation to the original sample, which refers to the percentage of the extracted correct information that accounts for all the recommended information in the sample. Diversity refers to the similarity between recommended lists. In other words, the overlap between the user's recommended lists defines the overall diversity. Coverage is the ratio of the number of items a system recommends to all users to the total number of items. Let *Ttest* represent all the recommended information in the sample. *Treco* represents all the information recommended by the recommender system for the user. That is, the recommended list length. *TP* represents the user's favourite information. Precision(P), Recall(R), Diversity(D) and Coverage(C) are defined as follows:

$$P = \frac{TP}{Treco} \quad (11)$$

$$R = \frac{TP}{Ttest} \quad (12)$$

$$D = \frac{2}{Ttest(Ttest-1)} \sum_{i,j \in U, i \neq j} \frac{|Treco_i \cap Treco_j|}{|Treco_i|} \quad (13)$$

$$C = \frac{N_d(Treco)}{Ttest} \quad (14)$$

The MAE measures the accuracy of the forecast by the deviation between the user's predicted ratings and the user's actual ratings. The smaller the value of MAE, the

smaller the difference between the predicted ratings and the actual ratings. This means that the forecast is more accurate and recommend higher quality. Set N to the number of items rated by the users in the test set. $P=\{i|P_i, i=1,2, \dots, k\}$ represents the actual ratings of the user. $Q=\{i|Q_i, i=1,2, \dots, k\}$ represents the user's predicted ratings, The MAE is defined as:

$$MAE = \frac{\sum_{i=1}^N |P_i - Q_i|}{N} \quad (15)$$

RMSE is a frequently used measure of the differences between values predicted by a model or an estimator and the values actually observed. RMSE can well reflect the precision of the measurement. Set N to the number of items rated by the users in the test set. $P=\{i|P_i, i=1,2, \dots, k\}$ represents the actual ratings of the user. $Q=\{i|Q_i, i=1,2, \dots, k\}$ represents the user's predicted ratings. The RMSE is defined as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (P_i - Q_i)^2}{n}} \quad (16)$$

5 Simulation Experiment and Result Analysis

In order to verify the effectiveness of the proposed method, we conducted a series of experimental analysis using the *MovieLens* small-scale data set provided by the *Grouplens* Working Group. The dataset contains 100 thousand sets of ratings from 943 independent users on 1682 movies, and the rating is an integer from 1 to 5. In the experiment, establish the matrix of ratings for the data set firstly. Secondly, the 100 thousand ratings are divided into 80 thousand training set and 20 thousand test set. Then, the simulation experiment and contrast analysis are carried out based on the user-based collaborative filtering algorithm, collaborative filtering recommendation based on Trust Model and collaborative filtering recommendation based on Trust Model with Fused Similar Factor.

5.1 Complexity Analysis

The time complexity of computing the similar factor, the direct trust degree, the degree of harmonic similarity, the degree of harmonic trust and the harmonic weight are all $O(n^2)$. However, time complexity of the overall adjustment of trust degree is $O(n^3)$, because it has to calculate the number of times users have made recommendations for other users.

5.2 Determine the Parameter of the Similar Factor

In the experimental modeling process, we firstly need to determine the parameter α in the formula (3) to solve the similar factor. The paper [9] pointed out that MAE will appear the minimum when α ranges from 1.0 to 1.5. Therefore, in order to determine the specific value of α in [1.0, 1.5], and the following experimental setup was

made: Set the number of neighbors in the model to 5, 10, 15, 20, 25, 30, 40, 60, 80, 100, and set the correct recommended threshold $e = 0.5$. According to the formula (8), we can calculate the degree of harmonic similarity, and the average values of MAE and RMSE corresponding to different neighbors are obtained at different values of α . The results are shown in Table.2.

Table 2. The average value of MAE and RMSE.

α	MAE	RMSE
1.0	0.8733	1.2128
1.1	0.8739	1.2147
1.2	0.8737	1.2140
1.3	0.8757	1.2165
1.4	0.8750	1.2156
1.5	0.8734	1.2132

Draw Fig.2 according to Table.2. Fig.2(a) and Fig.2(b) show the error curves for MAE and RMSE respectively. According to Fig.2 (a) and Fig.2 (b) can clearly see that when the α value of 1.0, MAE and RMSE are the minimum. Therefore, the value of α is set to 1.0.

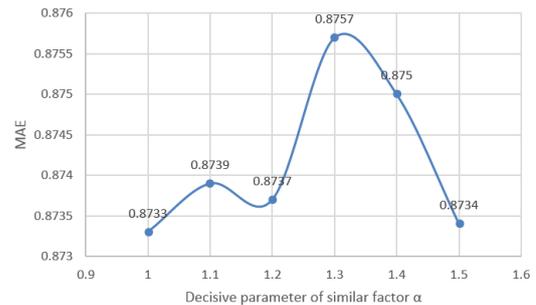


Fig. 2(a). The average value of MAE.

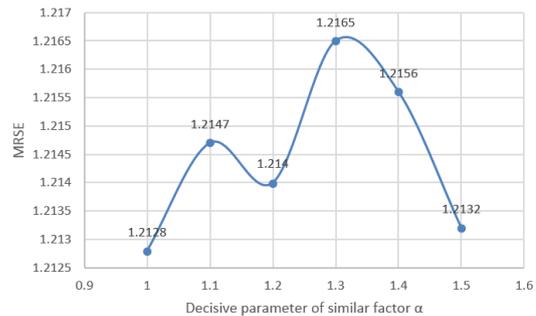


Fig. 2(b). The average value of RMSE.

5.3 Comparative Experiment

This paper compares collaborative filtering recommendation based on Trust Model with Fused Similar Factor (TMFSF) with collaborative filtering recommendation based on Trust Model (TM) and the User-Based collaborative filtering algorithm (UBCF). Precision, Recall, Diversity and Coverage are used in this article to measure the performance of the above algorithms. The experimental results are as follows:

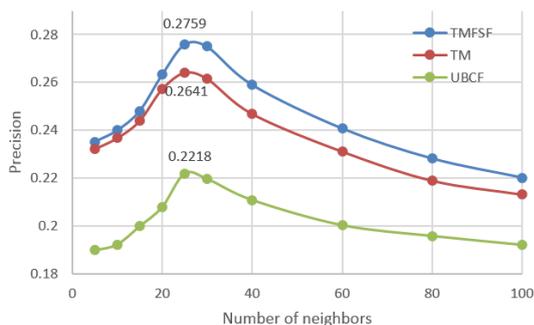


Fig. 3. Comparison of Precision.

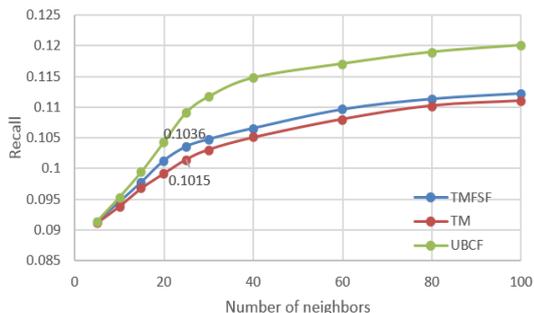


Fig. 4. Comparison of Recall.

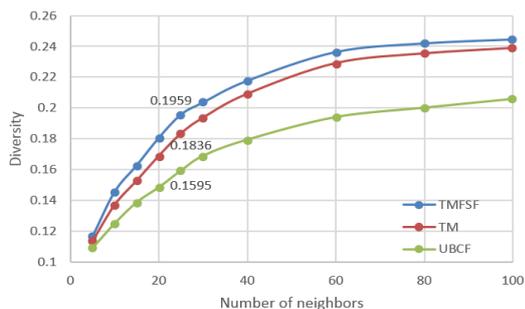


Fig. 5. Comparison of Diversity.

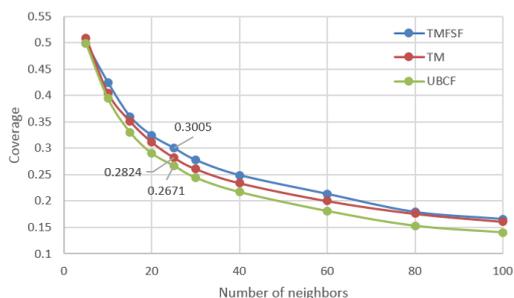


Fig. 6. Comparison of Coverage.

According to Fig.3, we can see, when the number of neighbors at 25, the Precision to reach the maximum. The TMFSF collaborative filtering algorithm is nearly 1.18% higher than the TM collaborative filtering algorithm. And it is 5.4% higher than the UBCF algorithm. The TMFSF collaborative filtering algorithm is superior to the UBCF algorithm and the TM algorithm.

As shown in Fig.4, the Recall of the TMFSF collaborative filtering algorithm is gradually increased as the number of neighbors increases. When the number of neighbors is 25, the TMFSF collaborative filtering algorithm is 0.21% higher than the TM collaborative filtering algorithm

Fig.5 is a comparison of the Diversity among the three groups. The TMFSF collaborative filtering algorithm is 1.23% higher than the TM collaborative filtering algorithm when the neighbor is 25. And it is 3.64% higher than the UBCF algorithm. This means that the TMFSF collaborative filtering algorithm can provide users with more choices.

Fig.6 is a comparison of the Coverage among the three groups. The TMFSF collaborative filtering algorithm is 1.81% higher than the TM collaborative filtering algorithm when the neighbor is 25. And it is 3.34% higher than the UBCF algorithm. Therefore, the TMFSF collaborative filtering algorithm can improve the user's satisfaction to some extent compared with the TM collaborative filtering algorithm.

The experimental results show that the TMFSF collaborative filtering algorithm in this paper has higher performance than the TM collaborative filtering algorithm, and its Precision, Recall, Diversity and Coverage have higher performance. Therefore, it can obtain High recommended quality.

6 Conclusions

In this paper, the concept of similar factor is introduced as an add-on to the collaborative filtering algorithm, which is based on the trust model. In calculating the user similarity, the impact of the number of common ratings between users is taken in to account. Similar factor is used to revise the similarity to make the similarity more accurate. The adjusted similarity is combined with the trust model to get the harmonic weights. Harmonic weights are combined with traditional collaborative filtering to produce higher quality recommendations. Consequently, we proved that the proposed algorithm Collaborative Filtering Recommendation Based on Trust Model with Fused Similar Factor is better than the traditional algorithm in Precision, Recall, Diversity and Coverage.

References

1. Wu, Xiaokun, B. Cheng, and J. Chen. "Collaborative Filtering Service Recommendation Based on a Novel Similarity Computation Method." *IEEE Transactions on Services Computing* **10**, (2017), 352-365
2. Chen. Hao, Z. Li, and W. Hu. An improved collaborative recommendation algorithm based on optimized user similarity. *Journal of Supercomputing* **72**, (2015), 2565-2578
3. Bobadilla, Jesús, F. Ortega, and A. Hernando. A collaborative filtering similarity measure based on singularities. *Information Processing & Management* **48**, (2012), 204-217
4. T Huang, R Huang. Improved Collaborative Filtering Recommendation Algorithm, *Computer Science*, **43**, (2016), 400-403
5. BK Patra, R Launonen, V Ollikainen, S Nandi. A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data, *Knowledge-Based Systems*, **82**, (2015), 163-177

6. G Guo,J Zhang,N Yorke-Smith.Leveraging multiviews of trust and similarity to enhance clustering-based recommender systems, *Knowledge-Based Systems*, **74**, (2015), 14-27
7. P Chen, C Chen. Incorporating Association Rules for Collaborative Filtering Recommendation Algorithm, *Journal of Chinese Computer Systems*, **37**, (2016), 287-292
8. Peng Zhenlian, Wang Jian, He Keqing, Tang Mingdong. A Requirements Elicitation Approach Based on Feature Model and Collaborative Filtering. *Journal of Computer Research and Development*, **53**, (2016), 2055-2066
9. GAO Quan-li, GAO Ling etc. Weighted Collaborative Filtering Algorithm with Fused Impact Facto [J]. *Computer Engineering*, **40**, (2014), 39-41
10. LU Kun, XIE Ling, LI Ming-chu. Research on Implied-trust Aware Collaborative Filtering Recommendation Algorithm [J]. *Journal of Chinese Computer Systems*, **37**, (2016), 241-243
11. GUO Yan-hong, DENG Gui-shi, LUO Chun-yu. Collaborative Filtering Recommendation Algorithm Based on Factor of Trust [J]. *Computer Engineering*, **34**, (2008), 1-3
12. WANG Jing, YIN Jian etc. Collaborative Filtering Recommendation Algorithm Based on Co-ratings and Similarity Weight [J]. *Computer Science*, **37**, (2010), 99-104
13. XU Xiang-yu, LIU Jian-ming. Collaborative Filtering Recommendation Algorithm Based on Multi-level Item Similarity [J]. *Computer Science*, **34**, (2016), 262-265
14. XIA Pingping , SHUAI Jianmei. Collaborative Filtering Algorithm Based on Similarity Extension and Interest Degree Scaling [J]. *Computer Engineering*, **42**, (2016), 199-202
15. WEN Jun-hao, SHU Shan. Improved Collaborative Filtering Recommendation Algorithm of Similarity Measure [J]. *Computer Science*, **41**, (2014), 68-71
16. C.Ziegler, S.M. McNee, J. A. Konstan et al. Improving recommendation lists through topic diversification. In: *Proceedings of the 14th International Conference on World Wide Web*. Chiba: ACM, (2005), 22-32
17. Ge M, Delgado-Battenfeld,Jannach D. Beyond accuracy: Evaluating recommender systems by coverage and serendipity. In: *RecSys (2010): the 2010 ACM conference on Recommender systems*. Barcelona:ACM,2010,257-260
18. Ge, Mouzhi, C. Delgado-Battenfeld, and D. Jannach. Beyond accuracy: evaluating recommender systems by coverage and serendipity. *Recsys*, (2010),257