International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 8 Issue: 5
DOI: https://doi.org/10.17762/ijritcc.v8i5.5382
***Article Received**: 04 January 2020 **Revised**: 12 March 2020 **Accepted**: 15 April 2020 **Publication**: 31 May 2020*

Research on User Interest Expression and Recommendation Service based on Three-dimensional Relationship of Users and Items

Dasong Sun
Office of Educational Administration, Nanjing University of Finance & Economics, Nanjing, China, 210023, Email: nuistsun@126.com

Shuqing Li
Corresponding author
Department of Information Management and Information System, College of Information Engineering, Nanjing University of Finance & Economics, Nanjing, China, 210023, Tel: +86-139-129-64688, Email leeshuqing@gmail.com

Wenjing Yan
College of Information Engineering, Nanjing University of Finance & Economics, Nanjing, China, 210023, Email 844569289@qq.com

Fusen Jiao
College of Information Engineering, Nanjing University of Finance & Economics, Nanjing, China, 210023, Email 569604824@qq.com

Junpeng Chen
College of Information Engineering, Nanjing University of Finance & Economics, Nanjing, China, 210023, Email luckjp@163.com

Abstract: The existing recommendation algorithms often rely heavily on the original score information in the user rating matrix. However, the user's rating of items does not fully reflect the user's real interest. Therefore, the key to improve the existing recommendation system algorithm effectively is to eliminate the influence of these unfavorable factors and the accuracy of the recommendation algorithm can be improved by correcting the original user rating information reasonably. This paper makes a comprehensive theoretical analysis and method design from three aspects: the quality of the item, the memory function of the user and the influence of the social friends trusted by the user on the user's rating. Based on these methods, this paper finally proposes a collaborative filtering recommendation algorithm (FixCF) based on user rating modification. Using data sets such as Movielens, Epinions and Flixster, the data sets are divided into five representative subsets, and the experimental demonstration is carried out. FixCF and classical collaborative filtering algorithms, existing matrix decomposition-based algorithms and trust network-based inference are compared. The experimental results show that the accuracy and coverage of FixCF have been improved under many experimental conditions.

Key words: time series; commodity genres; interest trend; recommendation algorithm; rating modification

ACKNOWLEDGMENTS:
This study was funded by Major Program of Natural Science Research in Colleges and Universities of Jiangsu Province of China 2019(grant number 19KJA510011)

1 INTRODUCTION

According to the China Internet Network Information Center, as of December 2018, the number of Internet users in China was 829 million, and the number of new Internet users was 56.53 million. The Internet penetration rate reached 59.6%, an increase of 3.8 percentage points from the end of 2017 [1]. In this era of information technology and the explosive growth of the Internet, people have entered the era of information overload. It is very difficult for users to find their own points of interest from the vast amount of information. For information producers, how to make their information stand out from the attention of users is also a complicated matter. At this time, the recommender system came into being. The task of the recommender system is to help users find information that is valuable to them and that they are interested in.

Amazon and NetFlix are both active participants and promoters of the recommender system. In China, we can also see that more and more Internet companies are promoting the application of recommendation systems, such as Taobao, Tencent and Netease. The personalized recommender system has penetrated into all aspects of people's food, clothing, housing and transportation, which has brought great convenience to people's lives. At the same time, it has also promoted the growth of consumption and promoted economic development. The recommendation algorithm is the core and critical part of the overall recommender system, which largely determines the performance of recommender system. Collaborative filtering has become the most widely used recommendation algorithm in the industry because of its strong versatility, not requiring too much expertise in the corresponding data collection, simple engineering implementation, and excellent recommendation effect [2].

At present, the mainstream collaborative filtering recommendation techniques fall into two categories: one is a neighborhood-based algorithm, such as a user-based collaborative filtering recommendation algorithm [3] and an item-based collaborative filtering recommendation algorithm [4], the other is based on model algorithms such as SVD [5], SVD++ [6], PMF [7], Local Low-Rank Matrix Approximation [8], etc. However, collaborative filtering also has shortcomings such as data sparsity, cold start and scalability [9], which often reduces the recommendation quality of the recommender system. The main research content of this paper is to modify user's score in rating matrix mainly from three dimensions: the influence of the quality of the item on the user's score, the influence of the user's memory on the user's score, and the influence of the user's social friend on the user's score. The purpose is to eliminate the influence of external factors on user ratings, to discover the user's true preference for the item, to improve the recommendation effect of the recommendation system, and to improve the user experience finally.

2 RELATED WORKS

There are many studies on the three factors of item quality, user memory and user trust.

Regarding the research on the quality of articles, the literature proposes the ItemRank algorithm based on the link analysis method, iteratively extracts the IR value of the item according to the algorithm, and recommends the high IR value to the user[10]. Document proposes a variant of PageRank to identify important nodes in the link and give a specific sort result[11]. Similarly, this link analysis method is not friendly to new items and is not actually used as a way to assess the quality of items. A number of text analysis methods using article reviews have also been proposed in to characterize the quality of articles[12][13][14].

Regarding the research on user memory, the literature proposed a recommendation model based on Ebbinghaus's forgetting curve, and introduced a time-window-based data weighting measurement method[15]. Literature proposed an improved model to adapt to changes in user interest, introducing time information to calculate Euclidean distance, adding current time for recommendation, and improving the timeliness of recommendation[16]. Borenstein proposed that the user's score will be affected by the order of the items to be graded, and a classifier that ignores the scoring order is proposed to improve the quality of the recommendation. Although the model is very simple, it is a good demonstration of the anchoring effect[17].

Regarding the research of social friends trusted by users, the literature proposed a trust-based singular value decomposition method (TrustSVD), which incorporates explicit information and implicit information into the model, and extends the SVD++ algorithm[18]. The recommended accuracy greatly increases the computational complexity. The Trust Matrix Decomposition Model (TrustMF) decomposes the scoring matrix and the trust matrix separately, and mixes the model to make recommendations, which effectively improves the accuracy of the recommendation[19]. Literature proposed SocialMF, which can effectively alleviate the cold start problem of recommended users by decomposing the trust matrix and merging the decomposed features[20].

However, it still needs further research to synthesize these methods and explore a more reasonable combination in the field of recommender system application, and give the overall relationship between algorithms and comprehensive design. This is also the main purpose of this paper.

3 IMPACT ANALYSIS FROM THREE DIMENSIONS OF USERS AND ITEMS

In recommender system, the relationship of users and items can be embodied in three important dimensions: the relationship between users, the relationship between items, and the relationship between users and items. The relationship between different dimensions will have a certain impact on the existing user rating behavior.

For example, in the relationship between items, we can use the characteristics between items to measure the quality of items. This quality will have a direct impact on user ratings, so that user ratings do not necessarily fully reflect the degree of fit between items and user interest, on the contrary, it only reflects the user's acceptance of the quality of items. Similarly, for user relationships, widespread social connections can also have a positive impact on user ratings. For the relationship between users and items, the historical scoring behavior also has a direct effect on the current user ratings.

These three dimensions and the corresponding analysis of related problems constitute the main research topics of this paper, which is shown in Figure 1:

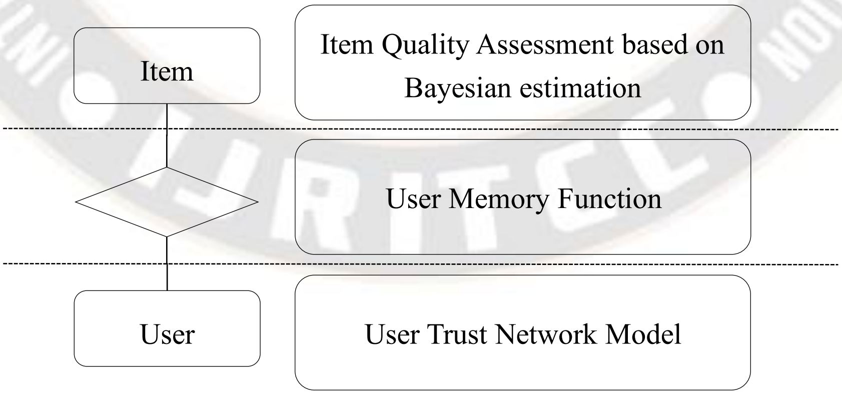

Figure 1. Three dimensions and corresponding research topic in this paper

***Article Received**: 04 January 2020 **Revised**: 12 March 2020 **Accepted**: 15 April 2020 **Publication**: 31 May 2020*

3.1 Item Quality Assessment based on Bayesian estimation

Common item quality assessment methods include mean method, link analysis method, and item review sentiment analysis. The average score of the user's rating of the item can reflect the overall situation of the user's rating of the item, and is also an important part of the user's collaborative filtering recommendation algorithm, but for items with less scoring, the average score is usually less able to reflect the real part of the item, and the use of the averaging method to assess the quality of items is highly vulnerable to external attacks. The use of link analysis as a method of evaluating the quality of an item also has problems such as low performance of the algorithm, vulnerable to control by attackers, and lack of effective weighting for new items. The algorithm of sentiment analysis based on item reviews is also very complicated and time-consuming, and the deficiency of item reviews often leads to large calculation errors.

Consider the following two situations:

(1) Movie A: 1 score record, with an average score of 5.

(2) Movie B: 50 score records, with an average score of 4.5.

One of the most intuitive ideas is to set a threshold K, which is the lower limit of the total score. Only the score number of movie above K can be calculated for average score, and these average scores will be ranked firstly. Those movies whose total score is less than K rank below the above results. However, this method has the following problems:

(1) How to select the threshold K.

(2) If there is only one movie with a total score greater than K in all movies, and its score is very low, then it is ranked first, it also may come to a conclusion contrary to the facts.

A more reasonable idea is that if you want to compare the quality of two movies, you should at least invite the same audience to watch and score. Bayesian estimation takes into account that there may not be enough data to estimate by calculation of the mean, and the influence of all other observed data is fully taken into account in this method. Using the Bayesian formula, we do not need to directly calculate a finite number of estimates, instead we calculate the probability distribution of the known values and then use this probability distribution to obtain the estimates.

The formula for IMDB to calculate TOP250[21] movies is shown in Formula 1:

$$\mathrm{WR} = \frac{V}{V+\mathrm{M}} \times R + \frac{M}{V+M} \times \mathrm{A} \quad 1)$$

Here R represents the average score of the movie, V represents the frequency of movie, M represents the minimum frequency of all movies, and A represents the average score of all movies. Drawing on this idea, the Bayesian estimation based item quality evaluation is proposed and shown in Formula 2:

$$r = \frac{C \times m + \sum ratings}{C+N} \quad 2)$$

Here m represents the a priori of the score (generally equal to the overall average score), C represents the confidence level of the prior (equal to the number of scores), and N represents the frequency of movie.

Going back to the beginning example, assuming m=3, C=5, the quality of movie A is:

$$r_A = \frac{5 \times 3 + 5 \times 1}{5+1} = 3.3$$

And the quality of movie B is:

$$r_B = \frac{5 \times 3 + 4.5 \times 50}{5+50} = 4.36$$

The estimated value of movie B is larger than the estimated value of movie A.

3.2 User Memory Function

The behavior of online users often does not manifest itself as a simple Markov process. The probability of behavior occurring due to long memory is higher than that of random behavior, which is consistent with a power law distribution. The behavior of online users is a complex process, and user's future behavior will be affected by past memory. Therefore, the user's scoring behavior on the items they consume is also affected by this memory effect, and users will score similar items according to their memory. The resulting interference will have a great negative impact on the recommendation effect of the recommender system. This is what researchers often call anchor effect.

A user may have multiple segments of memory, and the habits of scoring in different segments are different. We can define a memory segment of the current user from a score that is continuously higher than the average score of the user's score until it is lower than the average score of the user's score. Similarly, a memory segment can be defined from a score that is continuously lower than the average score of the user's score until it is higher than the average score of the user's score. Here P denotes the number of the user's score in a memory segment. For example, P = 5 means that the user gives five consecutive ratings higher (or lower) than his average score.

In order to assess the impact of user memory, we use a correlation coefficient formula [22]. The user's memory factor M is as shown in Formula 3:

$$M=\frac{1}{L-1}\sum_{i=1}^{L-1}\frac{(r_i-m_1)(r_{i+1}-m2)}{\sigma_1\sigma_2} \qquad 3)$$

Which r_i represents the i-th score of the user's scoring sequence, L represents the total number of scores in the user's memory segment, m_1, σ_1 representing the mean and standard deviation of $\{r_1,r_2,\ldots,r_{k-1}\}$, respectively, m_2, σ_2 representing the mean and standard deviation of $\{r_2,r_3,\ldots,r_k\}$. The value of M ranges from -1 to 1. Positive number indicates that users will continue to give high scores after high scores or low scores after low scores. On the contrary, negative number indicates that users will give low scores after high scores, or continue to give high scores after low scores. The larger the absolute value is, the more obvious this memory effect is.

3.3 User Trust Network Model

The trust relationship between users is usually represented as a directed graph and the trust model often includes four key definitions: the definition of trust, the measure of trust, the spread of trust, and the aggregation of trust. In the traditional recommender systems, trust between users can affect users' ratings, on the contrary, ratings can also reflect the degree of trust between users. The higher the trust between users, the more they will affect the user's rating. This paper uses a weight between [0, 1] to indicate the degree of trust between users, 0 means no trust, 1 means full trust. As shown in Figure 2, u1 has a degree of trust of 0.5 for u5 and a degree of trust of u2 of 0.6. Both u2 and u5 are social friends of u1 and have a direct trust relationship.

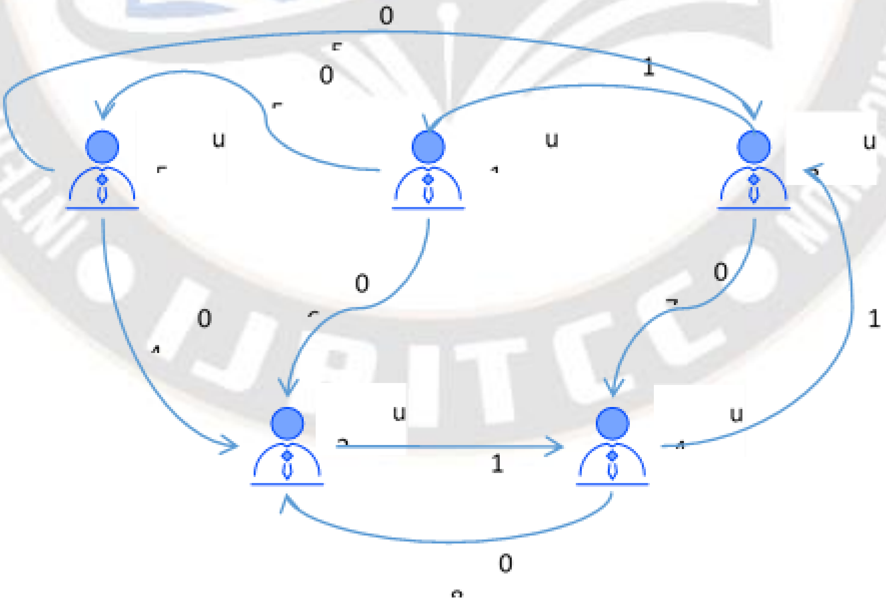

Figure 2. User Trust Network

There is not only direct trust between users, but also indirect trust. As shown in Figure 2, there is no direct trust relationship between u3 and u2, but there are paths of u3->u1->u2 and u3->u4->u2, and u3 and u2 form an indirect trust relationship, similar to friends in daily life. A friend of a friend may still be a friend.

If there is a direct path between users, the trust value is the current weight. If there is no direct path between users, similar to the one in [23], this paper uses weight spread to characterize the decrease in trust between users as the propagation of trust. The more nodes on the minimum distance between two user nodes, the lower the user trust. Taking Figure 2 as an example, there are two paths from u5 to u3: u5->u3 and u5->u2->u4->u3. Since there is a direct path between u5 and u3, u5 has a trust of 0.5 for u3. The path between u3 and u2 is u3->u1->u2 and u3->u4->u2, and there is no direct path. At this time, the two trust degrees are: 1*0.6=0.6 and 0.7*0.8=0.56. Trust degree exists on the multi-path in the trust network. If there is no direct trust relationship between the two users, this paper selects the indirect trust relationship with the largest trust value as the trust degree between the two users. To define a trust network G(V,E,T), the formula for calculating the trust between users u and v is as shown in Formula 4:

$$T(u,v) = T(u,i) \times T(i,v), i \in N_u \qquad 4)$$

Here N_u represents the set of neighbors of user u.

The pseudo code of trust network computing trust algorithm is given here.

```
Algorithm: Trust Network Computing Trust
Input: trust matrix
Output: Trust T
Process:
For user u, for i in test centralized user u rating items
Output each set of users U scored i from the training set
    While U is not empty
        for u1 in U
            If u and u1 have direct paths
        Add the trust values of u and u1 to the trust list
            Else if u and u1 are reachable and trust is greater than threshold (0.1)
Add trust to the list
    The maximum value in the output trust list is T
```

If two users have higher trust between each other, the two users are more similar. For example, we can calculate the similarity between any node u in the user trust network and all other nodes with the distant of nodes ranging from 1 to 6, respectively. Finally, by averaging the similarity at each distance, we can see that the trust between users decreases, and the similarity between users becomes lower. It also proves that in the trust network, as the distance (number of node) increases, the trust between users decreases continuously, and the similarity between users also decreases. This also proves that the higher the trust among users, the more similar the users are, and the greater the impact on user ratings. Then for some recommendation algorithms based on user similarity, the degree of user trust will have a more significant effect on the prediction method of user rating.

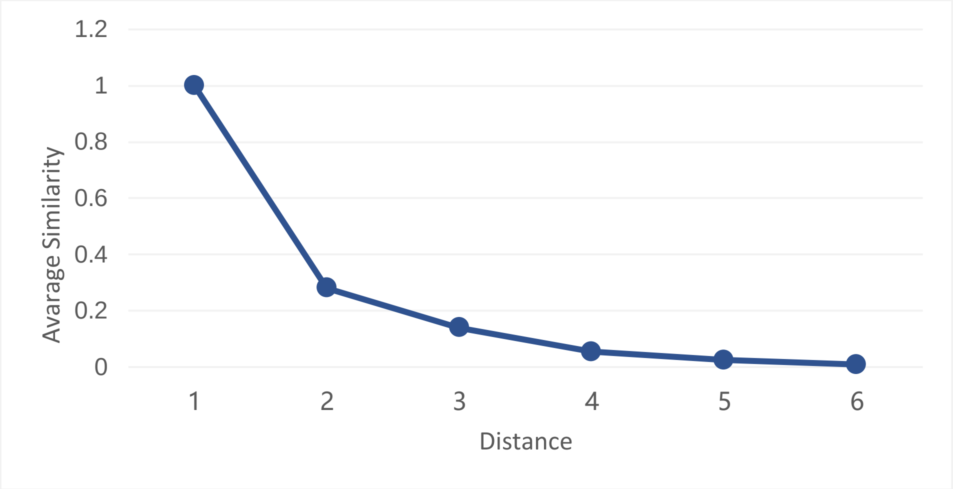

Figure 3. The relationship between user trust and user similarity

(Taking the Epinions data set as an example)

4 COLLABORATIVE FILTERING RECOMMENDATION ALGORITHM BASED ON USER RATING MODIFICATION

The most critical step in the collaborative filtering algorithm is the choice of neighbors, and this step usually uses the rating matrix to calculate the similarity for collecting neighbors. In this paper, a score modification algorithm is designed to eliminate the influence of these possible impact mentioned above. Then the modified rating matrix is used to recalculate the similarity and select the nearest neighbors that are more similar to the user's interest in order to improve the accuracy of the recommendation.

Based on the above research, the score modification formula is given in Formula 5:

$$r'_{ui} = \frac{(r_{ui} - MQ_i)}{r_{ui}} \times \bar{r}_i + \frac{(r_{ui} - T\bar{r}_u)}{r_{ui}} \times \bar{r}_u + \alpha \qquad 5)$$

Which $\bar{r}_u$ represents the average score of the user, r_{ui} representing the user's score on the item i, M representing the user's memory factor, Q_i representing the quality of the item i, T indicating the degree of trust, $\bar{r}_i$ indicating the average score of the item i, α indicating a modified offset to prevent over fitting. For example, if the user scored 5 points for one movie, the quality of this movie was 4 points, the average score of the movie was 3.8 points, the average score of the user was 4 points, the user memory factor was 0.4, and the user trust was 0.8. When set α to 0.3, the user's score can be modified as 4.32.

After modifying the rating matrix, we introduced the revised rating matrix into the user-based collaborative filtering to calculate the similarity, and predict the score to give a Top-n recommendation list. The calculation formula for predicting user score based on the collaborative score recommendation algorithm based on user score correction.

The basic step of the collaborative filtering recommendation algorithm FixCF based on user score modification can be divided into three modules, namely the input of the algorithm, the construction of the model and the generation of the recommendation. The basic process is as follows:

Step 1: Using the user's rating information, bring into the Formula 2, calculate the quality Q of each item.

Step 2: Arrange each user's ratings in chronological order, and divide each memory segment according to the algorithm, and bring into Formula 3 to calculate the memory factor M.

Step 3: Construct the trust network model by using the user's trust matrix, construct the trust propagation model, and bring in the Formula 4 to calculate the trust degree T.

Step 4: Using the previously calculated item quality Q, user memory factor M, user trust degree T, with the Formula 5, to calculate the modified ratings.

Step 5: Feed collaborative filtering algorithm with the modified ratings, to calculate the similarity between users, and predict ratings.

5 EXPERIMENT

This chapter will verify the feasibility of the FixCF algorithm through experiments and analyze the advantages of the FixCF algorithm through experimental comparison. Specifically, the main task is to complete the following experiments:

(1) Compared with the film score of IMDB, the authoritative scoring website of the film industry, the modification method based on Bayesian estimation proposed in this paper has better effect than the link analysis method and the average score method.

(2) Design experiments to observe the effects of different similarity and similarity thresholds on the accuracy of FixCF, and choose the appropriate similarity algorithm and similarity threshold.

(3) Design the experiment, verify the influence of the change of the score correction offset value on the FixCF accuracy, and select the most appropriate score correction offset.

(4) Divide the data set into five representative subsets, and on these subsets, FixCF and a series of state-of-the-art recommendation algorithms, including recommendation algorithms based on trust networks and recommendation algorithms based only on ratings are compared to observe the performance of FixCF in terms of accuracy and coverage.

The dataset used in this paper is the three datasets of Epinions, Movielens, and Flixster. To evaluate the performance of FixCF algorithm, the Flixster and Epinions data sets are divided into five different subsets:

(1) Cold-start users: A collection of users who scored items less than 20 times.

(2) Cold-start items: A collection of items with a user rating of less than 20 times.

(3) Heavy-rating users: A collection of users who have scored items more than 50 times.

(4) Struggle users: A collection of users who scored more than 20 items and whose standard deviation of scores was greater than 1.5.

(5) Biased items: Collection of items with a standard deviation of more than 1.5

The data set is divided into these five sets in order to verify the effectiveness of the algorithm in a variety of situations. The cold start user and the cold start item are divided to test the performance of the algorithm in the case of less user or item rating data. The division of heavy users is to test the performance of the algorithm in the case of more scoring data. Deviation of users and controversial items is to test the performance of the algorithm in the case of a large score span. The basic information of the Epinions data set and Flixster data set after partitioning is shown in Table 1 and Table 2:

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 8 Issue: 5
DOI: https://doi.org/10.17762/ijritcc.v8i5.5382
***Article Received**: 04 January 2020 **Revised**: 12 March 2020 **Accepted**: 15 April 2020 **Publication**: 31 May 2020*

Table 1. The basic information of the Epinions data set

Dividing set	User/Item Count	Number of scores
Cold-start users	50730	672640
Cold-start items	200760	2639400
Heavy-rating users	36201	463015
Struggle users	5445	591665
Biased items	17842	383652

Table 2. The basic information of the Flixster data set

Dividing set	User/Item Count	Number of scores
Cold-start users	57629	167880
Cold-start items	6597	123617
Heavy-rating users	30034	7927065
Struggle users	2896	128435
Biased items	7538	272211

In this paper, MAE (Mean Absolute Error), RMSE (Root Mean Square Error) and Coverage are used as indicators to evaluate the effectiveness of the algorithm. 20% of the data set is randomly divided into test sets, and the remaining 80% as a training set.

5.1 Analysis Experiment of the Quality of Items

Firstly, this paper selects the Movielens data set to verify the item quality assessment method based on Bayesian estimation. We use the Movielens dataset to experiment with method validity, and the scores are ranked in the top 10, top 50, top 100, top 150, top 200, and top 250. These results and the TOP250 of the IMDB website are compared. At the same time, we compare the effectiveness of this method with the mean method and representative algorithm of link analysis HITS. The hit rate is defined as shown in Formula 6:

$$P = \frac{Q}{I} \quad 6)$$

Q is the top 250 movie obtained by the item quality evaluation algorithm, and I represents the movie acquired by IMDB TOP 250.

In experimental results, of the top 10 movies in HITS, two are the same as IMDB. The hit rate is 0.2. However, Bayesian method shows that there are seven coincidences with IMDB in the top ten movies and the hit rate of Bayesian method is 0.7. The comparative analysis of the experiment hit rate is shown in Figure 4:

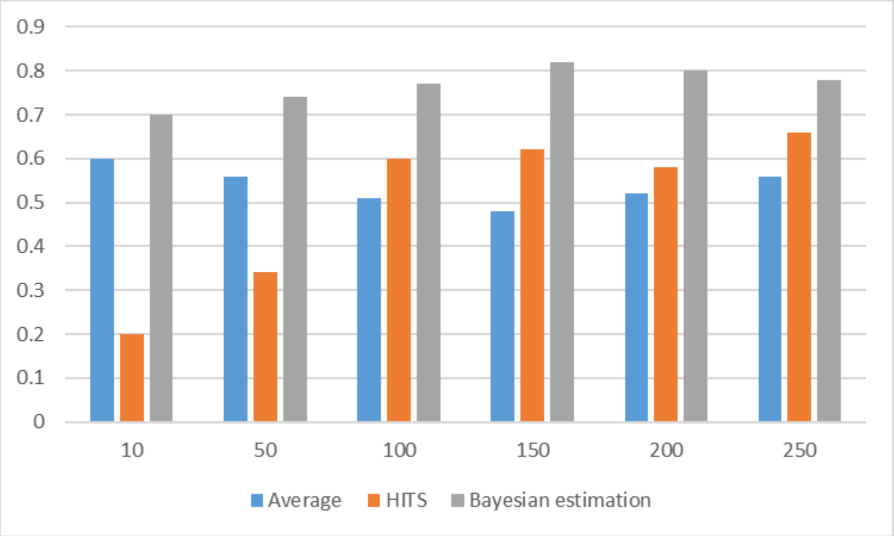

Figure 4. Comparison of item quality experiments

As can be seen from Figure 4, the results calculated by the item quality assessment method based on Bayesian estimation, the Top10, Top50, Top100, Top150, Top200 and Top250 always have the highest hit rate. In addition, the results obtained in the Bayesian-based method are quite close to the rankings of the Top250 on the IMDB official website. Therefore, it can be proved that the method for evaluating the quality of articles proposed in this paper has better experimental results than the common method of mean value and link analysis, indicating that the method is more effective.

5.2 The Effect of the Score Correction Offset Value on the Performance of the Recommendation Algorithm

In this paper, α is the bias of user score modification. This experiment takes Epinions dataset and Flixster dataset as examples to observe the impact of changes in values on the accuracy of recommendation results. The experimental results are shown in Figure 5

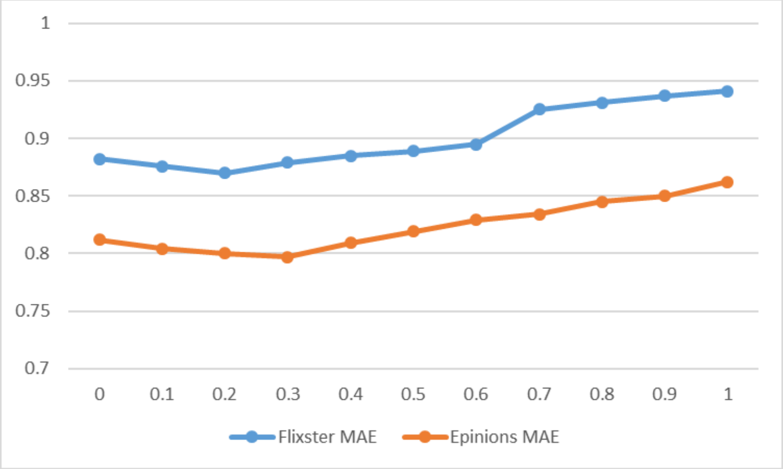

Figure 5. The effect of α on accuracy of recommendation

It can be seen from the experimental results that with the increase of α, the accuracy of the algorithm is firstly improved, and then decreased. The value of the recommendation is 0.3, and the accuracy of the recommendation result is the highest. Therefore, the value in the user score modification is 0.3 in our experiment.

5.3 Comparison between FixCF and Other Recommended Algorithms

The most important step in the collaborative filtering algorithm is the choice of neighbors. Firstly, experiments are used to verify the effect of threshold selection of similarity on the accuracy and coverage of collaborative filtering algorithms. The experimental results are shown in Figure 6 and Figure 7. It can be seen from the figure that the higher the similarity threshold is, the more similar the user interest is, and the higher the recommended accuracy is. However, at this time, the coverage of the recommended results is also reduced, and the long tail products cannot be effectively found. In addition, the case where the similarity is 1, means that the user's interest is completely similar, which is also a very ideal situation, and the real world is often difficult to achieve. Based on the above situation, the similarity threshold selected in this paper is 0.7. In the next section, we will experiment to compare the effects of different similarities on the accuracy of the recommendations.

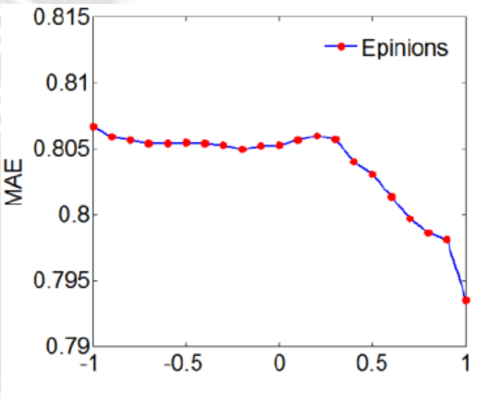

Figure 6. Similarity threshold on MAE

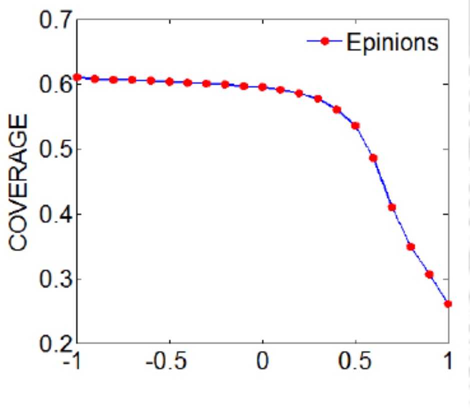

Figure 7. Similarity threshold on coverage

Based on the above experimental results, Cosine similarity is selected as the similarity metric of the collaborative filtering recommendation algorithm based on user score correction proposed in this paper.

In our experiments, the FixCF proposed in this paper is compared with a series of state-of-the-art recommendation algorithms, including recommendation algorithms based on trust networks and recommendation algorithms based only on scores. The experimental comparison recommendation algorithms based on the trust network include TrustSVD, TrustWalker, SocialMF and SocialRec. The experimental comparison recommendation algorithms based only on scoring include UserCF, ItemCF, PMF, SVD++, Local Low-Rank Matrix Approximation.

Table 2 shows the result on each subset of Epinions. Comparing the experimental data, Table 3 gives a comparison of the experimental data on each subset of the Flixster. The experimental results show that FixCF has the highest accuracy in most cases, and the average percentage improvement of accuracy ranges from 2.3% to 15%.

Table 2. Experimental comparisons on the Epinions dataset

Data set	Metrix	Recommendation algorithm						Average improvement
		UserCF	ItemCF	PMF	SVD++	LLRMA	FixCF	
Full data	MAE	0.887	0.883	0.872	0.841	0.852	0.825	4.8%
	RMSE	1.284	1.203	1.186	1.104	1.151	1.053	11.8%
Cold-start users	MAE	0.944	0.940	0.929	0.871	0.902	0.847	7.7%
	RMSE	1.383	1.322	1.279	1.127	1.254	1.082	15%
Cold-start items	MAE	0.866	0.850	0.847	0.831	0.840	0.825	2.6%
	RMSE	1.141	1.201	1.117	1.106	1.113	1.081	4.9%
Heavy-rating users	MAE	1.722	1.615	1.549	1.530	1.537	1.520	4.5%
	RMSE	2.050	1.988	1.896	1.849	1.873	1.862	9.8%
Struggle users	MAE	0.951	0.937	0.932	0.924	0.928	0.913	2.3%
	RMSE	1.309	1.240	1.213	1.198	1.206	1.171	5.0%
Biased items	MAE	1.615	1.536	1.549	1.542	1.541	1.502	3.5%
	RMSE	2.050	1.849	1.896	1.832	1.853	1.818	4.1%

Table 3. Experimental comparison on the Flixster dataset

data set	evaluatin index	Recommendation algorithm						Average improvement
		UserCF	ItemCF	PMF	SVD++	LLRMA	FixCF	
Full data	MAE	0.921	0.912	0.898	0.807	0.880	0.761	13.8%
	RMSE	1.119	1.116	1.052	0.987	1.033	0.964	9.2%
Cold-start users	MAE	0.957	0.950	0.818	0.789	0.802	0.797	7.7%
	RMSE	1.139	1.082	1.045	0.975	1.018	0.974	7.3%

Cold-start items	MAE	1.113	1.107	1.084	1.015	1.029	1.001	6.4%
	RMSE	1.449	1.350	1.300	1.259	1.267	1.250	5.7%
Heavy-rating users	MAE	0.929	0.889	0.879	0.837	0.865	0.834	5.2%
	RMSE	1.133	1.053	1.052	1.028	1.048	1.023	3.7%
Struggle users	MAE	1.594	1.528	1.525	1.412	1.516	1.415	6.6%
	RMSE	2.125	1.901	1.864	1.698	1.852	1.617	14.3%
Biased items	MAE	1.429	1.390	1.388	1.342	1.352	1.338	3.1%
	RMSE	1.919	1.881	1.825	1.606	1.737	1.612	10.1%

The comparative experimental of trust-based algorithms results in Table 4 show that FixCF is the best in terms of accuracy, with an average increase of 4.3%-11.8%.

Table 4. Comparison of FixCF and Trust-based algorithms

data set	evaluatin index	Recommendation algorithm					Average improvement
		TrustSVD	TrustWalker	SocialMF	SocialRec	FixCF	
Epinions	MAE	0.840	0.892	0.851	0.867	0.825	4.3%
	RMSE	1.092	1.124	1.096	1.101	1.053	4.6%
Flixster	MAE	0.793	0.855	0.824	0.846	0.761	8.3%
	RMSE	0.976	1.190	1.068	1.137	0.964	11.8%

The comparative experiments of coverage is shown in Figure 8. It can be seen that FixCF has a very high coverage rate in both Flixster and Epinions datasets. The reason why FixCF algorithm is lower than a few trust-based recommendation algorithms is that considering the time complexity of the experiment, FixCF algorithm has a slight limitation on the propagation of trust networks, which affects the coverage of the algorithm to a certain extent. The graph also proves that trust-based recommendation algorithm has a higher coverage rate than score-based recommendation algorithm, and can better mine long-tailed goods.

DOI: https://doi.org/10.17762/ijritcc.v8i5.5382
***Article Received**: 04 January 2020 **Revised**: 12 March 2020 **Accepted**: 15 April 2020 **Publication**: 31 May 2020*

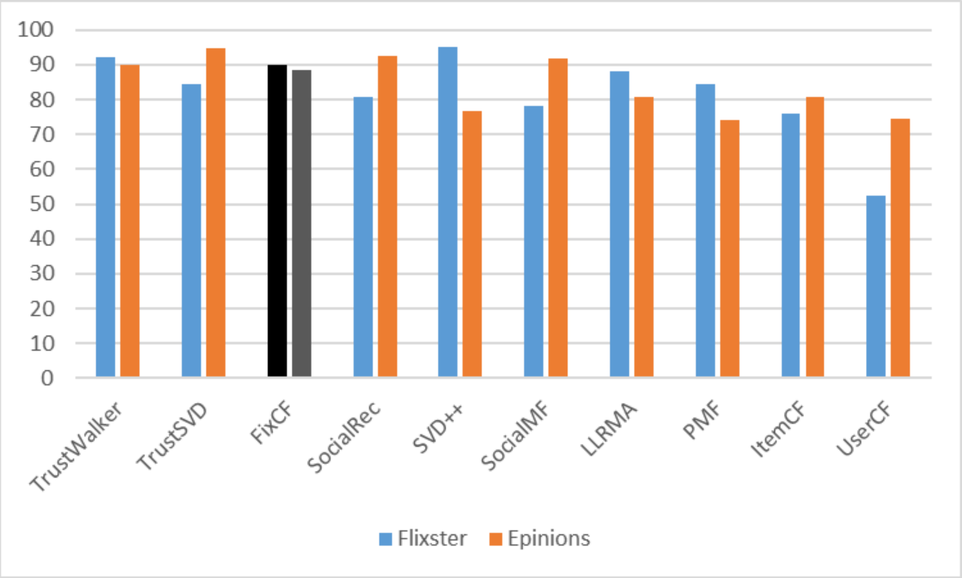

Figure 8. Comparison of coverage experiments

6 CONCLUSION

The user's rating information is usually the key point for the recommendation system to understand the user's interest characteristics, and usually the user's rating is often the result of multiple factors. This paper presents a novel recommending collaborative filtering algorithm focusing on correcting the scoring matrix based on the analysis of the relation of users and items.

The collaborative filtering based on user score correction proposed in this paper can greatly improve the accuracy of collaborative filtering recommendation. Using the user's social information can also alleviate the "cold start" problem to some extent. However, there are still some places worthy of further study. For example, how to further use the temporal information of user rating to explore the phenomenon of interest drift, and combine with content-based recommendation algorithm, to further improve the relevant algorithm design ideas and improve efficiency.

References

[1] China Internet Network Information Center. (2019). The 43rd Statistical Report on Internet Development in China. Beijing: China Internet Network Information Center.

[2] Xu Hailing, Wu Wei, Li Xiaodong. (2009). Comparative Study of Internet Recommendation Systems. Journal of Software, 20(2): 350-362.

[3] Deng Ailin, Zhu Yangyong, Shi Bole. (2003). Collaborative Filtering Recommendation Algorithm Based on Project Scoring Prediction. Journal of Software, 14(9): 1621-1628.

[4] Linden G, Smith B, York J. (2003). Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Computing, 7(1): 76-80.

[5] Li Gai, Li Lei. (2011). Collaborative filtering algorithm based on matrix decomposition. Computer Engineering and Applications, 47(30): 4-7.

[6] Koren Y. (2008). Factorization meets the neighborhood : a multifaceted collaborative filtering model. Proceedings of the 14th ACM SIGKDD International Conference of Knowledge Discovery and Data Mining, ACM Press.

[7] Salakhutdinov R, Mnih A. (2007). Probabilistic Matrix Factorization. International Conference on Neural Information Processing Systems.

[8] Lee J, Kim S, Lebanon G, et al. (2013). Local low-rank matrix approximation. International conference on machine learning, 82-90.

[9] Koohi H , Kiani K. (2017). A new method to find neighbor users that improves the performance of Collaborative Filtering. Expert Systems with Applications, 83:30-39.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 8 Issue: 5
DOI: https://doi.org/10.17762/ijritcc.v8i5.5382
***Article Received**: 04 January 2020 **Revised**: 12 March 2020 **Accepted**: 15 April 2020 **Publication**: 31 May 2020*

[10] Gori M, Pucci A, Roma V, et al. (2007). ItemRank: A Random-Walk Based Scoring Algorithm for Recommender Engines. IJCAI, 7: 2766-2771.

[11] Kang J, Wang M, Cao N, et al. (2018). AURORA: Auditing PageRank on Large Graphs, IEEE International Conference on Big Data (Big Data). IEEE, 713-722.

[12] Yang Fan. (2018). Online comment cluster analysis and recommendation based on LDA topic model. Dalian University of Technology.

[13] Liu Ying. (2015). Personalized Product Recommendation System Based on User Comments. Beijing University of Posts and Telecommunications.

[14] Feng X, Zeng Y, Xu Y. (2018). Recommendation Algorithm for Federated User Reviews and Item Reviews. Proceedings of the International Conference on Artificial Intelligence and Virtual Reality. ACM, 97-103.

[15] Yu, H., & Li, Z. (2010, October). A collaborative filtering method based on the forgetting curve. In 2010 International Conference on Web Information Systems and Mining, IEEE, Vol. 1, pp. 183-187.

[16] Hu Weijian, Teng Fei, Li Lingfang. (2016). Improved collaborative filtering algorithm adapted to user interest changes. Journal of Computer Applications, 36(8): 2087-2091.

[17] http://gregborenstein.com/assets/order-matters.pdf

[18] Guo G, Jie Z, Yorke-Smith N. (2015). TrustSVD: collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. Twenty-ninth AAAI Conference on Artificial Intelligence,123-129.

[19] LYang B, Lei Y, Liu J, et al. (2017). Social collaborative filtering by trust. IEEE transactions on pattern analysis and machine intelligence, 39(8): 1633-1647.

[20] Jamali M , Ester M. (2010). A matrix factorization technique with trust propagation for recommendation in social networks. Proceedings of the 2010 ACM Conference on Recommender Systems, RecSys 2010, Barcelona, Spain, September 26-30.

[21] https://www.imdb.com/chart/top

[22] Goh K I, Barabási A L. (2008). Burstiness and memory in complex systems. EPL (Europhysics Letters), 81(4): 48002.

[23] Podobnik B, Stanley H E. (2008). Detrended cross-correlation analysis: a new method for analyzing two nonstationary time series. Physical Review Letters, 100(8): 084102.

[24] Hu Xiangpei, Yin Jin. (2013). A Review of Research on Trust Transfer Model. Journal of Southeast University(Philosophy and Social Sciences), (4): 46-51.

[25] https://grouplens.org/datasets/movielens/

[26] http://socialcomputing.asu.edu/datasets/Flixster