

A Trust-Based Prediction Approach for Recommendation System

Peng Wang, Haiping Huang(✉), Jie Zhu, and Lingtao Qi

Nanjing University of Posts and Telecommunications, Nanjing, Jiangsu, China
hhp@njupt.edu.cn

Abstract. The recommendation system has been widely used in e-commerce, but still suffers from data sparsity and cold-start problems. This paper combines the user trust relationship with the collaborative filtering recommendation system and puts forward the recommendation approach based on trust delivery (TDR), in order to solve the above two problems. Through calculating the quantifying trust values between users, the prediction score of an unrated item can be figured out to achieve effective recommendation. Compared with other recommendation algorithms, TDR achieves better performance on standard Mean Absolute Error (*MAE*) and Coverage.

Keywords: Recommendation systems · Trust relationship · Collaborative filter

1 Introduction

In e-commerce applications, the main task of the recommendation system is how to determine which items satisfy the users' demands or interests and then recommend them to users, in order to reduce searching time and improves users' experience [1]. At present, the research on recommendation system mainly focuses on the analyzing and modeling of the user's interest preference and historical behavior data, so as to accurately predict the future behaviors of users. The most representative achievement, such as the collaborative filtering recommendation system, have been gradually applied in the business fields.

However, the classic collaborative filtering recommendation algorithm still faces great challenges, especially for the following two issues:

1. Data sparsity problem: with the rapid growth of users and items in recommendation system, the number and variety of recommended items increases fast, and the overlap of similarity selections between users will be less and less, which causes the data is becoming sparse.
2. User cold-start problem: for some new users in the recommendation system, there is little historical data about them because they have not rated to any item, thus it is difficult for the system to give an accurate personalized recommendation.

Based on the above problems, this paper designs a new recommendation approach. Based on the trust delivery relationship between users, even with the sparse score matrix, it can still calculate the user prediction ratings for given items which are not

© Springer International Publishing AG, part of Springer Nature 2018
A. Yang et al. (Eds.): SERVICES 2018, LNCS 10975, pp. 157–164, 2018.
https://doi.org/10.1007/978-3-319-94472-2_12

rated. And meanwhile, it can guarantee desirable performance on the Mean Absolute Error (*MAE*) and Coverage.

This paper is organized as follows: Sect. 2 describes the related work and background knowledge; Sect. 3 introduces the recommendation approach based on trust delivery (TDR); the experimental results and analysis are presented in Sect. 4; Sect. 5 summarizes the whole paper.

2 Related Work and Background

2.1 Related Work

The basic idea of collaborative filtering recommendation [2] is to analyze the similarity between users or items based on historical data, and then to use similarity to predict the current user's preference of items. They can be divided into user-based recommendation and item-based recommendation [3]. Aiming at the existing problems such as data-sparsity and cold-start, many researchers have provided improved methods. Zhao et al. proposed several new similarity definition methods for different requirements; however, there is no obvious improvement on data-sparsity and cold-start problems [4]. Hwang et al. proposed a collaborative filtering algorithm using uncertain nearest neighbors, which alleviates the sparsity of data to some extent; its defect lies in it is hard to balance the recommended results between user group and item group [5]. Shambour et al. combined the trust relationship between users with collaborative filtering algorithm, and then proposed TSF algorithm to solve both the data-sparsity and cold-start problems [6]. Similarly, O'Donovan et al. [7] divided the user trust relationship into explicit trust relationship and implicit trust one, for the purpose of overcoming the above two problems. However, [6, 7] failed in giving the precise definitions of the degree of trust among users without considerations of combining the number of items rated by two users with the average score of other items.

2.2 Background Knowledge

The 'trust' in this paper is the expectation that whether the target users can provide valuable rating ability to other users [8]. Trust is a measurement of the degree of trust between users, which has a range of [a, b]. For example, in the range of [0,1], 0 indicates distrust instead 1 means complete trust. The trust relationship usually has the following characteristics: (1) Asymmetry; (2) Transitivity; (3) Dynamic: with the change of time and interactions between users, the trust value may change accordingly.

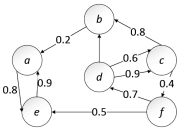


Fig. 1. User trust relationship graph

As shown in Fig. 1, the trust relationship between users can be represented by a directed-graph due to asymmetric trust relationship. User c trusts user b with the trust value 0.8, user b trusts user a the trust value 0.2, then user c is also trust user a.

3 A Trust-Based Delivery Recommendation Method

In this section, we first describe the recommendation approach based on trust delivery (TDR), and then give an example. Suppose that the original score matrix is R_{m*n}, it means that m users rate to n items, which can cause serious cold-start problem due to the matrix sparsity. The structure of TDR is shown in Fig. 2.

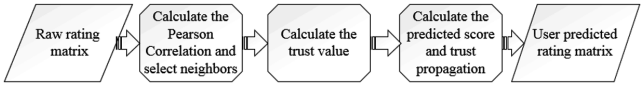


Fig. 2. Structure of recommendation approach based on trust delivery.

3.1 Calculate the Pearson Correlation Between Users

Pearson correlation is used to calculate the similarity between users, which is calculated by (1):

$$\rho_{a,b} = \frac{\sum_{i=1}^{I_{a,b}} \left(R_{a,i} - \bar{R}_a\right) \times \left(R_{b,i} - \bar{R}_b\right)}{\sqrt{\sum_{i=1}^{I_{a,b}} \left(R_{a,i} - \bar{R}_a\right)^2} \times \sqrt{\sum_{i=1}^{I_{a,b}} \left(R_{b,i} - \bar{R}_b\right)^2}} \tag{1}$$

where $R_{a,i}$ and $R_{b,i} \in [1, 5]$ represent the ratings of user a and user b on item i respectively, R_a and $\overline{\mathrm{R}}_b \in [1, 5]$ represent the average rating values of user a and user b on all items separately. $I_{a,b}$ is the set of co-rated items by both user a and user b. Based on (1), we can find two users with high similar similarity. However, when user a and user b have only one co-rated item, according to (1), the value of $\rho_{a,b}$ is 1. It is obviously unreasonable, so we need to add the correction parameter $\beta_{a,b}$, which is calculated by (2), where $|N_a|$ is the number of items rated by user a, $|N_b|$ is the number of items rated by user b, and $|N_{a,b}|$ is that have been commonly rated by a and b.

$$\beta_{a,b} = \frac{|N_{a,b}|}{|N_a| + |N_b| - |N_{a,b}|} \tag{2}$$

Therefore, Pearson Correlation between users is redefined as (3):

$$\rho'_{a,b} = \begin{cases} \rho_{a,b} \times \beta_{a,b}, & |N_{a,b}| \neq 0 \\ NONE, & |N_{a,b}| = 0 \end{cases} \tag{3}$$

3.2 Trust Calculation and Propagation

It takes the original score matrix as input and calculates the direct trust values between the selected user a and a group of k neighbor users. After calculating the direct trust value, the indirect trust values can be derived using the transitivity of trust. The calculation of direct trust value is shown as (4) and (5):

$$D_{a,b} = \frac{1}{2}(\delta(\bar{R}_a) + \delta(\bar{R}_b)) \times \beta_{a,b} \tag{4}$$

$$\delta(x) = \begin{cases} 1 - \frac{1}{x + |N_{a,b}|}, & |N_{a,b}| \neq 0 \\ NULL, & |N_{a,b}| \end{cases} \tag{5}$$

where R_a, $\overline{R}_b \in [1, 5]$ and $|N_{a,b}|$ denotes the number of items co-rated by user a and user b. If user a and user b have a lot of common rated items, then the value of $|N_{a,b}|$ will be larger, and the value of $\delta(x)$ will be larger, so the value of $D_{a,b}$ will be larger too. If the value of $|N_{a,b}|$ is 0, it means that user a and user b do not share the common rated items, and they have no direct trust relationship.

However, not all users have direct trust relationship, for example, user a and user c in Fig. 1. While relying on user b, indirect trust relationship can be explored between them. So, the trust propagation can be calculated by (6):

$$DP_{a,c} = Min\{D_{a,b}, D_{b,c}\} \times \delta(\bar{R}_a) \tag{6}$$

where $Min\{x,y\}$ represents the minimum value of x and y. Finally, based on (7), we can calculate the predicted scores of the non-rated items of user a.

$$PR_{a,i} = \frac{1}{1 + D_{a,b}} \times \bar{R}_a + \frac{D_{a,b}}{1 + D_{a,b}} \times \left(R_{b,i} - \bar{R}_b\right) \tag{7}$$

3.3 Example Analysis

Suppose that 4 users can rate 6 kinds of items, the value range is [1, 5], and the greater the score value is, the higher satisfaction the user have. The user's original scoring matrix is shown in Table 1. N indicates the user does not rate the item. It can be found that user U_1 is a cold start user, which means this user only rates few items.

Table 1. User original scoring matrix

	I_1	I_2	I_3	I_4	I_5	I_6
U_1	N	N	3	N	N	N
U_2	4	3	4	2	N	N
U_3	4	2	N	4	N	4
U_4	N	1	N	N	4	5

Table 2. User person correlation matrix

	U_1	U_2	U_3	U_4
U_1		0	*NONE*	*NONE*
U_2			0.03	0.17
U_3				0.38
U_4				

According to (1), (2) and (3), we can calculate the Pearson Correlation between users. The calculation results are shown in Table 2. Then, according to (4) and (5), users' trust values can be calculated, Table 1 shows that U_1 and U_3, U_1 and U_4 have no public ratings, which means there are no direct trust relationship between them. Therefore, the trust values between these two groups of users cannot be calculated directly, which can be marked with "*NULL*", as shown in Table 3.

Table 3. User direct trust value matrix

	U_1	U_2	U_3	U_4
U_1		0.19	*NULL*	*NULL*
U_2	0.19		0.5	0.13
U_3	*NULL*	0.5		0.3
U_4	*NULL*	0.13	0.3	

Table 4. User trust matrix

	U_1	U_2	U_3	U_4
U_1		0.19	0.14	0.1
U_2	0.19		0.5	0.13
U_3	0.16	0.5		0.3
U_4	0.09	0.13	0.3	

According to Table 3, U_1 can establish indirect trust relationship with U_3 and U_4 via U_2, and the specific indirect trust values can be calculated in (6). The complete user trust value matrix will be constructed, as shown in Table 4. Finally, based on the direct or indirect trust values of users, the scores of the users' unrated items are predicted by (7), as shown in Table 5.

Table 5. User predicted score matrix

	I_1	I_2	I_3	I_4	I_5	I_6
U_1	2.64	2.48		2.32	2.79	2.88
U_2					2.96	3.07
U_3			1.25		2.84	
U_4	2.62		1.89	2.62		

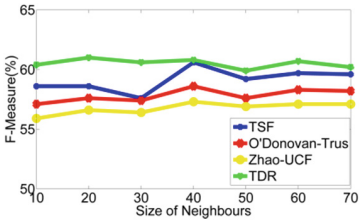


Fig. 6. *F-measure*-comparisons when size of neighbors varying.

4 Experimental Evaluation and Discussion

The MovieLens 100 K data set is selected as the rating databases of recommendation system, and this data set is provided by the GroupLens research team at the University of Minnesota U.S, which includes 100000 ratings records of 943 users to 1682 movies. Each rating record includes a user ID, an item ID, a value of rating in the interval [1, 5] and a TimeStamp.

4.1 Evaluation Metrics

The standard Mean Absolute Error (*MAE*) has been used to evaluate the quality of recommendations. *MAE* is the most widely used metric to measure the accuracy of

recommendations. Note that a lower *MAE* value represents a higher recommendation accuracy. The measurement for *MAE* can be given by (8):

$$MAE = \frac{\sum_{i=1}^{n} \left| RA_{a,i} - RP_{a,i} \right|}{n} \tag{8}$$

where $PA_{a,i}$ denotes actual rating, $RP_{a,i}$ represents system's predicted rating, and n denotes the number of available items.

Coverage is used to evaluate the ability of a specific recommendation system to provide recommendations, which is calculated by (9):

$$Cov = \frac{|N_p|}{n} \tag{9}$$

where $|N_p|$ denotes the number of items for which a prediction rating has been made, n denotes the number of available items.

Precision and *Recall* are usually used to evaluate the efficiency of recommendation system. *Precision* can be figured out with (10):

$$P_u = \frac{N_{tp}}{N_{tp} + N_{fp}} \tag{10}$$

where N_{tp} represents the number of items which are recommended and meanwhile appreciated by users; N_{fp} represents the number of items which are recommended while not approved by users.

Similarly, *Recall* can be defined by (11):

$$R_u = \frac{N_{tp}}{N_{tp} + N_{tn}} \tag{11}$$

Where N_{tn} represents the number of items which are not recommended while liked by users. In this paper, if the user's rating of item is greater than or equal to 3, that means the item is preferred by this user.

In general, the more comprehensive evaluation *F-Measure* can be obtained by weighting and averaging *Precision* and *Recall*, as shown in (12):

$$F - Measure = \frac{(\alpha^2 + 1) P_u \times R_u}{\alpha^2 (P_u + R_u)} \tag{12}$$

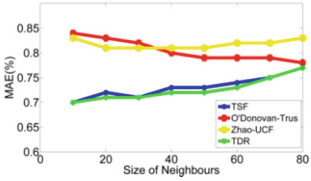
where the value of α is usually set to 1.

4.2 Experimental Results and Analysis

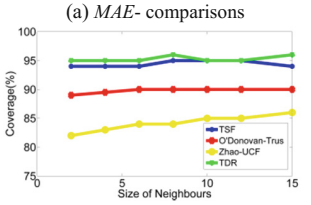
In this section, TDR will be compared with other typical recommendation algorithms to verify the performance of our proposal. The data sets come from MovieLens.

Zhao-UCF is a user-based collaborative filtering recommendation algorithm and its similarity is calculated using the Person correlation [4]. TSF algorithm [6] and O'Donovan-Trust algorithm [7] are based on trust relationship. These algorithms have been viewed in the related work.

We calculate the values of *MAE* of these recommendation algorithms by changing the number of user neighbors. Figure 3(a) shows that TDR has the highest recommended accuracy with any given number of neighbors. It owes to the rational utilization of user trust relationship, which has significantly improved the recommended accuracy, especially the number of neighbour is between 30 and 70. Figure 3(b) illustrates that TDR has the highest Coverage with any given number of neighbors. This is because TDR not only adopts the improved Pearson correlation, but also employs the user's trust relationship, which causes a significant improvement in recommendation coverage compared with other recommendation algorithms.

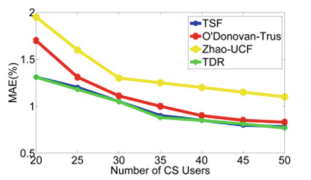


(a) *MAE*- comparisons

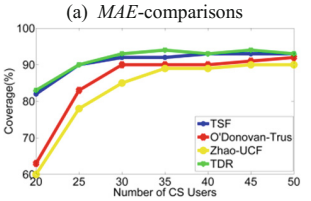


(b) Coverage-comparisons

Fig. 3. Comparisons when size of neighbors varying.

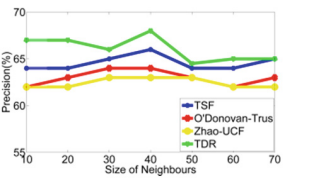


(a) *MAE*-comparisons

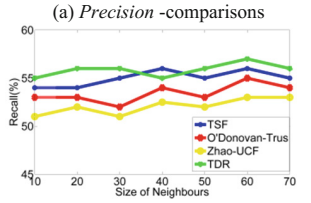


(b) Coverage-comparisons

Fig. 4. Comparisons when the number of CS users varying.



(a) *Precision* -comparisons



(b) *Recall*-comparisons

Fig. 5. Comparisons when size of neighbors varying.

MAE and Coverage of TDR are measured when the number of cold-start users varying, as shown in Fig. 4(a) and (b). It can be found that in comparison with other recommended algorithms, TDR achieves better performance on both *MAE* and Coverage. The experimental results show the effectiveness of TDR when dealing with the problem of cold-start users.

Figure 5(a) and (b) show the *Precision* and *Recall* rates of the TDR, respectively. Figure 5(a) illustrates that TDR keeps a high *Precision* rate in any given neighbors. Figure 5(b) shows that TDR also has a high *Recall* rate compared with other algorithms. The experimental results demonstrate the user satisfaction of TDR.

Figure 6 shows the *F-Measure* performance obtained by weighting and averaging the *Precision* and *Recall* rates. Both of them in Fig. 5(a) and (b) are higher than others

in any given number of neighbors, so in Fig. 6 the *F-Measure* of TDR has an advantage over other algorithms.

5 Conclusions

In view of data sparsity and cold-start problems in recommendation system, we design a novel recommendation approach based on trust delivery (TDR), which quantifies user trust relationship to calculate user's trust values and scores for unrated items. Through experimental comparisons with typical algorithms, TDR achieves more desirable performance in terms of recommendation accuracy and coverage, and meanwhile the trust transitive relationship between users can effectively solve the cold user problem and provide higher recommendation satisfaction.

Most of the existing methods focus on the trust relationship network among users. However, there is few research on the distrust relationship among users, which should become one of our future work. In addition, the improvement of similarity measure between items deserves further concern in order to alleviate the case of item cold-start and improve the quality of recommendation.

Acknowledgement. This work was supported by the National Natural Science Foundation of P. R. China (No. 61672297), the Key Research and Development Program of Jiangsu Province (Social Development Program, No. BE2017742).

References

1. Dakhel, A.M., Malazi, H.T., Mahdavi, M.: A social recommender system using item asymmetric correlation. Appl. Intell. **5**, 1–14 (2017)
2. Zheng, Z., Ma, H., Lyu, M.R.: A collaborative filtering based web service recommender system. In: IEEE International Conference on Web Services, Los Angeles, CA, USA, 6–10 July 2009, pp. 437–444. IEEE, July 2009
3. Zhao, Y.S., Liu, Y.P., Zeng, Q.A.: A weight-based item recommendation approach for electronic commerce systems. Electron. Commer. Res. **17**(2), 205–226 (2017)
4. Zhao, Z.D., Shang, M.S.: User-based collaborative-filtering recommendation algorithms on Hadoop. In: International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 9–10 January 2010, pp. 478–481. IEEE, January 2010
5. Hwang, C.-S., Chen, Y.-P.: Using trust in collaborative filtering recommendation. In: Okuno, H.G., Ali, M. (eds.) IEA/AIE 2007. LNCS (LNAI), vol. 4570, pp. 1052–1060. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73325-6_105
6. Shambour, Q., Lu, J.: A trust-semantic fusion-based recommendation approach for e-business applications. Decis. Support Syst. **54**(1), 768–780 (2012)
7. O'Donovan, J., Smyth, B.: Trust in recommender systems. In: International Conference on Intelligent User Interfaces, San Diego, California, USA, 10–13 January 2005, pp. 167–174. ACM, January 2005
8. Cao, G., Kuang, L.: Identifying core users based on trust relationships and interest similarity in recommender system. In: IEEE International Conference on Web Services, San Francisco, USA, 27 June–2 July 2016, pp. 284–291. IEEE, June–July 2016