

Implicit and Explicit Trust in Collaborative Filtering

Yuanxin Ouyang^{1,2(✉)}, Jingshuai Zhang^{1,2}, Weizhu Xie^{1,2},
Wenge Rong^{1,2}, and Zhang Xiong^{1,2}

¹ Engineering Research Center of Advanced Computer Application Technology,
Ministry of Education, Beihang University, Beijing 100191, China

² School of Computer Science and Engineering,
Beihang University, Beijing 100191, China

{oyyx,weizhushieh,w.rong,xiongz}@buaa.edu.cn, zhangjs_it@163.com

Abstract. Recommender Systems based on collaborative filtering could provide users with accurate recommendation. However, sometimes due to data sparsity and cold start of the input ratings matrix, this method could not find similar users accurately. In the past, researchers used implicit trust weight instead of the similarity weight to find similar users, to improve the quality of recommendation [17]. And they often ignore the role of explicit trust in the process of finding similar users. Therefore, in this paper, we explore the calculation of implicit trust and explicit trust. Then according to their role in the recommendation system, we propose a method that combined trust and similarity to get a better recommendation. At last, by experimenting on FilmTrust [5] data set which has the explicit trust matrix, the result showed that the method we proposed significantly improve the quality of recommendation, in addition, implicit trust and explicit trust have a positive effect on the quality of the results of recommendation.

Keywords: Recommender system · Collaborative filtering · Implicit trust · Explicit trust

1 Introduction

Collaborative filtering is one of the most successful method in the recommendation field. In 1990s, researchers began to pay attention on this field, and collaborative filtering also has promoted the development of the whole research of personalized recommendation. In 1992, Goldberg et al. was the first to apply collaborative filtering to the recommendation system, and collaborative filtering became the first algorithm in recommendation field. In the early twenty-first Century, the world's top researchers organized the first meeting of the Recommender System ACM in the field of personalized recommendation. After that, the field of recommendation system has been greatly developed. Until now, both in academic research and commercial application, personalized recommendation have been made many achievements, and are changing people's habit to make

people easily find the useful information for them in the age of information explosion.

However, there are still some problems such as data sparsity and cold start in the traditional collaborative filtering technology, which has seriously caused some negative influences on the quality of the results of recommendation. In the following research work, in order to improve the quality of recommendation, the researchers proposed many solutions, such as reducing the dimension of rating matrix [20], adding social relation in matrix operation [9], combining the item similarity and user similarity to find similar users. Recently, researchers have started to introduce machine learning and deep learning to recommendation system and some of them have got better results [18,21], but these methods are not mature, most of these methods cost a lot. Now take these problems from another point of view, we have considered the influence of interpersonal relationships on the recommendation system.

In our life, people often take the advice of friends who is trusted when they make decisions. Through these suggestions, the correctness of the decision will be greatly improved [1]. So in the Internet age, this relationship has become particularly important. In the traditional recommendation system, it often ignores the relationship, and the user is divided into a separate individual to consider. Therefore, no matter how we improve the recommendation, we cannot solve the actual problem of recommendation. Application of trust network among users in recommendation system, on the one hand can be combined with the users own characteristics and their neighbors and better mining user preferences and interest information [13], improve the quality of the recommendation system. On the other hand, through the analysis of the trust network can provide additional information, such as trust propagation, we can alleviate the problem of data sparsity and cold start problems in traditional recommendation system in a certain extent. So it is very necessary to study the trust network among the recommender systems.

At present, in the research of the application of trust network in the recommendation system, the explicit trust and the implicit trust are separately studied. The explicit trust is often used in collaborative filtering of matrix factorization [8,12,14]. And on the contrary, the implicit trust is used in collaborative filtering based on user [19]. In fact, the combination of the above two kinds of trust on the recommendation system also has an important significance, and it can better improve the quality of the recommendation.

In this paper, in order to solve the problem of data sparsity and cold start, we introduce a more abundant trust network included implicit trust and explicit trust to the recommendation system. We combine user similarity and trust as the new weight to find similar users and to recommend. Through several sets of control experiments, we found that either explicit or implicit trust could improve the recommendation results. And this proved the validity of trust network. Before the combination, we firstly discussed the calculation of implicit trust. Then because the trust is propagable, the method of trust propagation are developed. After

having more abundant social relations, we can find the similar users accurately and get more accurate rating prediction, and this model named TPKNN.

2 Related Work

In order to attain the information which is useful for us efficiently, the recommendation system is widely applied to recommend films, books and songs. Collaborative filtering is one of the most successful technologies that has been applied in recommendation field. It gets huge success in both e-commerce and radio website applications. There are two kinds of collaborative filtering, the recommendation method based on items and based on users. The collaborative filtering based on items generated in mathematical statistics and the collaborative filtering based on users came from habits of people, the latter is more explanatory.

Collaborative filtering based on users is one of the earliest proposed collaborative filtering method, and it has been recognized in various industry recommendations. The core of collaborative filtering is mining the relationship between users, between items and between user and item. From these historical data, we can find the users potential behavior rules and recommend new items to users. And this is exactly using the shopping habits of people. K nearest neighbor model is a typical representative of collaborative filtering, and its working principle is using historical data to find k neighbors who is the most similar to the target user, then use the neighbor set to predict the target user's preference for the item. This approach does not feature extraction for the users or items, so this method is suitable for the item which is not structured.

In 1996, for the first time, M. Blaze et al. proposed the concept of trust management in their study [2]. However their purpose is to solve the security problem of Internet service. After the application, the concept of trust was born and many researchers began to put the trust management mechanism apply to recommendation systems. Trust management mechanism in the application of recommendation method is becoming more and more mature after the formulation of the calculation and propagation of trust. Although there is no uniform method to get trust network, researchers are still finding a better method.

Paolo Massa et al. discussed the propagation mode of explicit trust [15], and John O'Donovan et al. proposed the calculation method of implicit trust [16]. Yang Bo et al. in the collaborative filtering based on matrix factorization discussed the relationship between trustee and truster in the recommendation, and gave a method to combine the front two [22], and Soude et al., in the matrix factorization, compared the difference to the result of recommendation between the explicit trust and the implicit trust [3]. Khosravani compared the different trust recommender models in collaborative filtering recommendation [11]. Recently, Zheng Yaoyao et al. had put the distrust into trust networks to form a new trust model [23]. And this model improved the quality of recommendation for the dataset which has the relationship of distrust. So in this paper, we proposed a new networks combined explicit trust and implicit on these basis to get better results of recommendation.

3 TPKNN: A Model Combined with Similarity and Trust

3.1 Basic Steps of Collaborative Filtering

The collaborative filtering based users is a method which is recommended by the rating similarity, is also the first proposed and widely used in the commercial recommendation system. The collaborative filtering based users need find the similar users firstly, then generates the rating prediction by the set of similar users. And the general process includes getting user similarity matrix, finding K neighbors and prediction [4].

Getting User Similarity Matrix

This step is mainly for data preprocessing. Firstly, user similarity refers to the degree of similarity between the two users in the evaluation of the item, and it is standard to select the neighbors. The original input data is rating matrix, and the column vector and row vector of the matrix represent the set of users who rates the item and the set of items which rated by the user, respectively. So we can use the method of calculating the similarity between the vectors to express the similarity between the users and the users.

At present, the score similarity calculation method can be divided into the following three categories Cosine Similarity, Adjusted Cosine Similarity, Pearson Correlation Coefficient [1]. Cosine Similarity is the most basic and simple method to get the rating similarity, which is expressed by the cosine of the angle between the two vectors. However, it doesn't take into account the different rating criteria from all kinds of users. So Adjusted Cosine Similarity takes a way to eliminate this difference, and Pearson Correlation Coefficient further optimizes the method of calculating the similarity between users. In order to get more accurate results, we use the Pearson Correlation Coefficient to get the user similarity matrix, the formula is as follows:

$$sim(i, j) = \frac{\sum_{t \in K_i^R \cap K_j^R} (r_{it} - \bar{r}_i) \bullet (r_{jt} - \bar{r}_j)}{\sqrt{\sum_{t \in K_i^R} (r_{it} - \bar{r}_i)^2} \bullet \sqrt{\sum_{t \in K_j^R} (r_{jt} - \bar{r}_j)^2}} \quad (1)$$

where K_i^R and K_j^R denote the set of items which rated by user i and user j , respectively, \bar{r}_i and \bar{r}_j denote the mean rating of user i and user j , respectively.

Finding K Neighbors

The K nearest neighbors of user is a set of K users who have the highest similarity with the target users. And they have similar interests with the target users. Although it is the easiest step, the selection of neighbors will also affect the quality of the recommended results. For the user u , we first get the similarity between user u and other users, and then sorted from largest to smallest, finally select the front K users as the neighbor users. The number of nearest neighbor K determines the recommended range of changes. Whether the K value is too large or too small, the accuracy of the recommendation will not be guaranteed. So in this paper, we will take several groups of experiments to observe the effect

of different K on the recommended result, and the number of K is 5, 10, 20, 30, 40, 50.

Prediction

Prediction would get the recommended result. After the user similarity matrix and neighbor users are obtained, we can predict the rating for users. Similar to the method of calculating the similarity, we also need take into account the different rating criteria from all kinds of users. And the formula is as follows:

$$\hat{r}_{ui} = \bar{r}_u + \frac{\sum_{v \in T_u \cap K_i^R} sim(u, v) \bullet (r_{vi} - \bar{r}_v)}{\sum_{v \in T_u \cap K_i^R} sim(u, v)} \tag{2}$$

where $sim(u, v)$ denotes the similarity between user u and user v , T_u denotes the neighbor users of user u , \bar{r}_u and \bar{r}_v denote the mean rating of user u and user v , respectively.

This recommendation method is widely applied in all kinds of fields because it is easy to interpret and it has better accuracy, we called it PKNN. However, data sparsity and cold start problems still exist, so in the following section, we introduce trust network to the collaborative filtering to solve these problems.

3.2 Trust Network

Trust network is a complex network composed of interpersonal relationships. From the way of acquisition, it is divided into explicit trust and implicit trust. Explicit trust can be obtained directly from the dataset, and it comes from the mutual evaluation between users. The value of explicit trust is generally 0 or 1, and 0 and 1 represent distrust and trust respectively, so when combined with the similarity we need a weight to change the explicit trust between 0 and 1.

However, up to now, the trust network is very sparse, so we need add implicit trust to supplement. Implicit trust is derived from the users historical rating matrix [7], and it can also mining the hidden trust relationship between users [16]. Since the correctness of the recommendation has indicated the difference of user preferences, the accuracy of the recommendation can be used as the standard to calculate the implicit trust between two users. When calculating the implicit trust of the user u and the user v , we put the user v as a single neighbor of the user u , then we use the formula (2) to calculate the prediction rating of user u on the item i , set \hat{r}_{ui} . And we use $Correct(i)$ to denote the set of items which is correct prediction, the formula is as follows:

$$Correct(i) = \{i | |\hat{r}_{ui} - r_{ui}| < e\} \tag{3}$$

where r_{ui} denotes the real rating to item i from user u , e is a fixed value that represents the range of allowable errors.

Let the $Total(i)$ denotes the total number of items which have been predicted, and the formula for calculating the implicit trust between user u and user v is as follows:

$$trust(u, v) = \frac{count(Correct(i))}{Total(i)} \tag{4}$$

We can get the implicit trust matrix by this method easily. In this formula, $count(Correct(i))$ is always less than $Total(i)$, in other word, the molecular is always less than the denominator, so the range of the implicit trust value is between 0 and 1, which is convenient to combine with the similarity.

However, the above method to calculate the trust network is not enough, the trust network is still sparse. To avoid the problem, we introduce the method of trust propagation to the trust network [6]. Moreover, the method of explicit trust propagation and implicit trust propagation is different.

In the process, we use $R(m, n)$ to show the path between the node m and node n , that is the path of trust. And $N_{R(m,n)}$ indicates that the number of nodes on the path. In the method of explicit trust propagation, let d denotes the maximum propagation distance, and the formula is as follows:

$$DT_{R(m,n)} = (d - N_{R(m,n)})/d \tag{5}$$

In the method of implicit trust propagation, there are two rules to calculate the indirect trust.

Rule 1: There is only one shortest path in the paths of node m to node n , the formula is as follows:

$$DT_{R(m,n)} = trust(m, a_1) \times trust(a_1, a_2) \times \dots \times trust(a_k, n) \tag{6}$$

where a_i denotes the node in the path of trust.

Rule 2: There are more than one shortest path in the paths of node m to node n , the formula is as follows:

$$DT_{R(m,n)} = \frac{\sum_i^k DT_{R_i}}{k} \tag{7}$$

where DT_{R_i} is got by the **Rule 1**.

At the last, because the attenuation of the trust propagation we set the maximum propagation distance d to 4 in our research. And this not only increases the speed of calculation but also does not reduce the accuracy of the recommendation.

3.3 The New Weight

According to the above method, we can get the new trust network, then combine trust and similarity to get a new weight for finding the neighbor users and predicting. When we combine one trust and similarity, the formula of the new weight is as follows:

$$W(u, v) = sim(u, v) * (1 - \alpha) + trust(u, v) * \alpha \tag{8}$$

where the $trust(u, v)$ denotes the explicit trust or implicit trust, α denotes the weight of the explicit or implicit trust. And when we combine two trust and similarity, the formula of the new weight is as follows:

$$W(u, v) = sim(u, v) * (1 - \alpha - \beta) + etrust(u, v) * \alpha + mtrust(u, v) * \beta \tag{9}$$

where $etrust(u, v)$ denotes the explicit trust, $mtrust(u, v)$ denotes the implicit trust, α denotes the weight of explicit trust, β denotes the weight of implicit trust.

After get the new weight, we can replace user similarity with the new weight, and the new formula of prediction is as follows:

$$\hat{r}_{ui} = \bar{r}_u + \frac{\sum_{v \in T_u \cap K_i^R} W(u, v) \bullet (r_{vi} - \bar{r}_v)}{\sum_{v \in T_u \cap K_i^R} W(u, v)} \quad (10)$$

where $W(u, v)$ denotes the new weight got from the above.

In this section, we first introduced the steps of collaborative filtering, and then introduced the trust network in detail and the calculation of getting explicit trust and implicit trust, finally get a new weight combined similarity and trust and put the new weight to the formula of prediction to get the final recommendation results. The above is the steps of TPKNN to recommend, and next we will experiment to verify the validity of this method.

4 Experiments and Validations

4.1 Description of Dataset

The dataset used in our experiments is from the FilmTrust website which uses trust in web-based social networks to create predictive movie recommendations. The FilmTrust dataset is a small dataset crawled from the entire FilmTrust website in June, 2011 [5]. And the date contains rating matrix and trust matrix, so we can get the explicit trust easily. In FilmTrust dataset, the range of rating id between 0.5 and 4, the value of explicit trust matrix is 0 or 1, representing the distrust or trust relationship, respectively. This dataset contains 1642 users and 2071 items, the total number of ratings is 35497 and the total number of trust is 1853. So the dataset is very sparse, and its density is 1.04% and 0.069% in terms of ratings and trust relations, respectively.

4.2 Experimental Setup

Evaluate metrics. In the experiment, we use the classical evaluation metrics to test the prediction quality of the proposed model, which is the mean absolute error (MAE) and the root mean square error (RMSE) [10].

The formula of MAE is as follows:

$$MAE = \frac{\sum_{i,j} |R_{i,j} - \hat{R}_{i,j}|}{N} \quad (11)$$

where $R_{i,j}$ represents the rating user i gives to item j , $\hat{R}_{i,j}$ denotes the rating user i gives to item j , and N is number of test dataset.

The formula of RMSE is as follows:

$$RMSE = \sqrt{\frac{\sum_{i,j} (R_{i,j} - \widehat{R}_{i,j})^2}{N}} \quad (12)$$

where $R_{i,j}$ represents the rating user i gives to item j , $\widehat{R}_{i,j}$ denotes the rating user i gives to item j , and N is number of test dataset.

Comparison Methods. In order to comparatively evaluate the performance of our proposed methods, we compare the following four methods, and the existing method is traditional recommender method which only use the user similarity. And the details are as follows:

PKNN: the traditional collaborative filtering recommendation which using Pearson Correlation Coefficient to get the user similarity matrix and set the user similarity as the weight for finding k neighbors and predicting [1].

ETPKNN: proposed in this paper which combined the user similarity and explicit trust as the new weight for finding k neighbors and predicting.

MTPKNN: proposed in this paper which combined the user similarity and implicit trust as the new weight for finding k neighbors and predicting.

TPKNN: proposed in this paper which combined the user similarity and all trust as the new weight for finding k neighbors and predicting.

Cross-Validation. In the model training and rating testing, we used cross validation to process the data set. In simple terms, cross validation divides the data set into k parts, and then selects the (a) parts as a training set, and the rest ($K-a$) parts as a test set. Because the FilmTrust data set is sparse, we use a 5-fold cross-validation. In each time we randomly divide the data into 5 parts, then select 1 part as the test set and the rest 4 parts as the training set. That is to say, there are 80 % of the data for the training set, 20 % of the data for the test set. We make a total of 5 times tests, and take the mean as the final result.

4.3 Results

In Eqs. 8 and 9, it can be seen that the weight of trust α and β is very important for recommendation, since it shows the proportion of user similarity, explicit trust and implicit trust, and only when we find the best value of α and β , we can get the best comparison results. So before comparing these models, we need first determine the optimal weight of trust. According to the experimental results, we first plot the change of MAE according to the weight of explicit and implicit trust, as shown in Fig. 1, where the number of neighbor is set to 20. And we can find that when the weight of explicit trust is 0.2 and the weight of implicit trust is 0.3, the MAE is minimum, which means the performance of recommendation is the best at that time.

So in the ETPKNN model, we set the weight to 0.2 and the weight set to 0.3 in the MTPKNN model. Similar to the above experiment, the weight of explicit

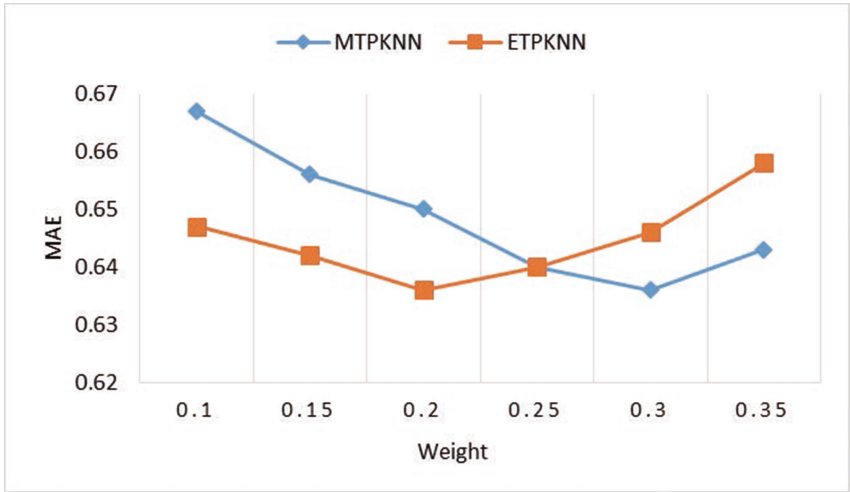


Fig. 1. Impact of weight

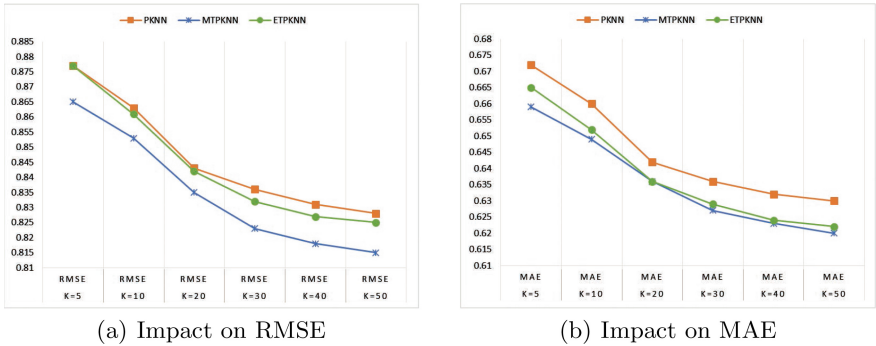


Fig. 2. The comparison of recommendation performance

trust and implicit trust are both set to 0.18 in the TPKNN model. We could find that the weight of user similarity is the highest no matter which model is, and the weight of implicit trust is higher than explicit trust in single model.

By comparing the performance of ETPKNN, MTPKNN and PKNN to verify the effectiveness of the trust network in the recommendation. The experimental results are shown in Fig. 2. It shows that the performance of ETPKNN and MTPKNN is better than PKNN, so it proves that our theory about trust network and the propagation trust is reasonable, and this combination can improve the accuracy of the recommendation. But on the other hand, although the performance of ETPKNN does not improve much, it costs the same time as PKNN. On the contrary, the model of MTPKNN improves the accuracy of recommendation obviously, but it needs a lot of time to calculate the implicit trust.

Table 1. Comparison result against baselines in term of MAE and RMSE

| The number of neighbor | Measure | PKNN | MTPKNN | ETPKNN | TPKNN |
|------------------------|---------|-------|--------|--------|-------|
| 5 | MAE | 0.672 | 0.659 | 0.665 | 0.653 |
| | RMSE | 0.877 | 0.865 | 0.877 | 0.863 |
| 10 | MAE | 0.66 | 0.649 | 0.652 | 0.646 |
| | RMSE | 0.863 | 0.853 | 0.861 | 0.854 |
| 20 | MAE | 0.642 | 0.636 | 0.636 | 0.632 |
| | RMSE | 0.843 | 0.835 | 0.842 | 0.836 |
| 30 | MAE | 0.636 | 0.627 | 0.629 | 0.624 |
| | RMSE | 0.836 | 0.823 | 0.832 | 0.824 |
| 40 | MAE | 0.632 | 0.623 | 0.624 | 0.62 |
| | RMSE | 0.831 | 0.818 | 0.827 | 0.82 |
| 50 | MAE | 0.63 | 0.62 | 0.622 | 0.617 |
| | RMSE | 0.828 | 0.815 | 0.825 | 0.817 |

After verifying the validity of the trust network, we do another experiment to verify the performance of the TPKNN model and the results are shown in Table 1. We can find that the TPKNN performs the best of all, no matter how much the number of neighbors. On one hand, the more neighbors there are, the better the quality of the recommendation is. Until the number of neighbors increased to 40, the change of MAE is not obvious, which indicates that the recommended model has contained almost users who can be trusted, the increase of neighbors would not improve the accuracy of the recommendation. On the other hand, the MAE of TPKNN with 5 neighbors is less than the MAE of PKNN with 10 neighbors. However, the time of PKNN is the sum of the time of the ETPKNN and MTPKNN, so when TPKNN obtains more accurate results, it sacrifices other resources at the same time.

5 Conclusion

In this paper, we proposed a model combined user similarity and the trust networks which combined explicit trust and implicit trust. In the model, we not only use the user-item rating matrix, but also use the user-user trust matrix to get social relations. The experiment on Filmtrust dataset shows that the model we proposed model performs better compared with traditional method, so the validity of trust network is proved. The model also confirmed the validity of the way to get explicit trust and implicit trust, and the possibility of combination between user similarity and trust. So it could be a direction of the future study.

The model proposed in this paper showed the superiority in experiment, but it's definitely not the best. So in the future work, there are still several challenges for the researchers. Firstly, making the way of trust propagation more reasonable would make the trust network more abundant, then the recommendation result

would be more accurate. Secondly, the way of combination between user similarity and trust could be more complex instead of a simple linear combination. Whats more, the success of the TPKNN model would encourage more people to explore the calculation of trust networks.

Acknowledgements. This work was partially supported by the National Natural Science Foundation of China (No. 61472021), SKLSDE project under Grant No. SKLSDE-2015ZX-17 and the Fundamental Research Funds for the Central Universities.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* **17**(6), 734–749 (2005)
2. Blaze, M., Feigenbaum, J., Lacy, J.: Decentralized trust management. In: 1996 IEEE Symposium on Security and Privacy, Proceedings, vol. 30, no. 1, pp. 164–173 (1996)
3. Fazeli, S., Loni, B., Bellogin, A., Drachsler, H., Sloep, P.: Implicit vs. explicit trust in social matrix factorization. In: Proceedings of the 8th ACM Conference on Recommender systems, pp. 317–320 (2014)
4. Ge, X., Liu, J., Qi, Q., Chen, Z.: A new prediction approach based on linear regression for collaborative filtering. In: Eighth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2011, Shanghai, China, 26–28 July 2011, pp. 2586–2590 (2011)
5. Golbeck, J.: Generating predictive movie recommendations from trust in social networks. In: 4th International Conference, Trust Management, iTrust 2006, Proceedings, Pisa, Italy, 16–19 May 2006, pp. 93–104 (2006)
6. Guha, R.V., Kumar, R., Raghavan, P., Tomkins, A.: Propagation of trust and distrust. In: Proceedings of the 13th International Conference on World Wide Web, pp. 403–412 (2004)
7. Guo, G., Zhang, J., Thalmann, D., Basu, A., Yorke-Smith, N.: From ratings to trust: an empirical study of implicit trust in recommender systems. In: Proceedings of 29th ACM International Symposium on Applied Computing, pp. 248–253 (2014)
8. Guo, G., Zhang, J., Yorke-Smith, N.: TrustSVD: collaborative filtering with both the explicit and implicit influence of user trust and of item ratings. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, Austin, Texas, USA, 25–30 January 2015, pp. 123–129 (2015)
9. Guo, G., Zhang, J., Yorke-Smith, N.: A novel recommendation model regularized with user trust and item ratings. *IEEE Trans. Knowl. Data Eng.* **28**(7), 1607–1620 (2016)
10. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.: Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.* **22**(1), 5–53 (2004)
11. Khosravani, A., Farshchian, M., Jalali, M.: Introduction of synthetic and non-synthetic trust recommender models in collaborative filtering. In: International Congress on Technology, Communication and Knowledge, pp. 1–9 (2014)
12. Koren, Y., Bell, R.M., Volinsky, C.: Matrix factorization techniques for recommender systems. *IEEE Comput.* **42**(8), 30–37 (2009)
13. Ma, H., King, I., Lyu, M.R.: Learning to recommend with social trust ensemble. In: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 203–210 (2009)

14. Ma, H., Yang, H., Lyu, M.R., King, I.: SoRec: social recommendation using probabilistic matrix factorization. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, pp. 931–940 (2008)
15. Massa, P., Avesani, P.: Trust-aware collaborative filtering for recommender systems. In: Meersman, R., Tari, Z. (eds.) OTM 2004. LNCS, vol. 3290, pp. 492–508. Springer, Heidelberg (2004). doi:[10.1007/978-3-540-30468-5_31](https://doi.org/10.1007/978-3-540-30468-5_31)
16. O'Donovan, J., Smyth, B.: Eliciting trust values from recommendation errors. In: Eighteenth International Florida Artificial Intelligence Research Society Conference, Clearwater Beach, Florida, USA, pp. 289–294 (2005)
17. O'Donovan, J., Smyth, B.: Trust in recommender systems. In: Proceedings of the 10th International Conference on Intelligent User Interfaces, pp. 167–174. ACM (2005)
18. Portugal, I., Alencar, P., Cowan, D.: The use of machine learning algorithms in recommender systems: a systematic review. arXiv preprint [arxiv:1511.05263](https://arxiv.org/abs/1511.05263) (2015)
19. Roy, F., Sarwar, S.M., Hasan, M.: User similarity computation for collaborative filtering using dynamic implicit trust. In: Khachay, M.Y., Konstantinova, N., Panchenko, A., Ignatov, D.I., Labunets, V.G. (eds.) International Conference on Analysis of Images, Social Networks and Texts, pp. 224–235. Springer, Heidelberg (2015)
20. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Application of dimensionality reduction in recommender system—a case study. Technical report, DTIC Document (2000)
21. Wang, H., Wang, N., Yeung, D.Y.: Collaborative deep learning for recommender systems. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1235–1244. ACM (2015)
22. Yang, B., Lei, Y., Liu, D., Liu, J.: Social collaborative filtering by trust. In: International Joint Conference on Artificial Intelligence, pp. 2747–2753 (2013)
23. Zheng, Y., Ouyang, Y., Rong, W., Xiong, Z.: Multi-faceted distrust aware recommendation. In: International Conference on Knowledge Science, Engineering and Management, pp. 435–446 (2015)