*International Journal of Intelligent Information and Management Science*
*ISSN: 2307-0692, Volume 8, Issue 2, April, 2019*

# Improved Collaborative Filtering Algorithm based on a Novel Similarity Measure for Recommender Systems

Yannian Chen, Fei Chu, Mingyuan Wang, Linjia Sun
Faculty of Computer Science and Technology, Anhui University, Hefei, 230601, China

**Abstract:** With the explosive growth of services and items on the internet, recommender systems have been widely used to recommend personalized items to users. Collaborative filtering (CF) is one of the most successful recommendation techniques used in recommender systems. Similarity computation is the critical step, which will significantly affect the predictive accuracy of CF. Traditional similarity measures, such as Pearson's correlation coefficient (PCC) and cosine similarity (COS), have inherent limitations on accuracy. And there are still many problems in the improved measures proposed recently in some papers. To address these problems, we fully consider different situations that may be encountered in the process of recommendation and propose a novel similarity measure. Based on the proposed similarity measure, we propose a new collaborative filtering algorithm to improve the accuracy of recommendation and meanwhile enhance the robustness of recommender systems. The results of experiments conducted on three real datasets prove that our CF algorithm achieves excellent performances.

**Keywords:** Recommender system; Similarity measure; Collaborative filtering; Predictive accuracy; RDCF

## 1. Introduction

The emergence of the e-commerce made many users enter the Internet where enormous amount of resources are provided for items they demanded. However, with the rapid development of computer and information technologies, the amount of data is increasing dramatically. It becomes difficult, time-consuming and ineffective for users to gain valuable information on the Internet. Thus, recommender systems emerged as tools to help users tackle the enormous amount of information they are confronted with Recommender systems do not require users to express their demands clearly, but predict their demands by analyzing their historical behaviors and then recommend personalized items to them[20]. In the big data era, recommender systems have become an effective solution to alleviate the "information overload" problem. Nowadays, recommender systems have been applied to recommend a variety of online items, such as online videos, online shopping, online services and online social networks. Recommendation techniques mainly include content-based recommendation, collaborative filtering (CF) recommendation and hybrid recommendation[19], in which CF techniques are more frequently adopted and often result in better predictive accuracy [10-11]. Differing from the content-based recommendation [21-22], CF doesn't consider the content of items or extract the features of content. Instead, it focuses on user ratings in the user-item matrix and intends to find the similarity between users or items.

CF mainly includes user-based CF[6] and item-based CF [4-5]. The idea of user-based CF is that users with similar historical ratings might have similar interest, so we can predict the missing ratings of a target user on a specific item according to the ratings of similar users on the given item. Thus, user-based CF calculates similarities between users to find the most similar ones as a neighborhood and recommends items based on the ratings of the neighborhood. Similar to user-based CF recommendation algorithm, item-based CF calculates the similarity between items and generates a neighborhood of a target item, and then predicts missing ratings based on the ratings of the neighborhood.

Similarity computation is a critical step of CF, which is usually conducted on a user-item matrix where each row represents the row rating vector of a specific user on all items and each column represents the column rating vector of a specific item given by all users. There are two similarity computation measures widely implemented in CF recommender systems: Pearson's correlation coefficient (PCC) [3-9] and cosine similarity (COS)[4]. These two traditional measures have some inherent limitations which result in their relatively low predictive accuracy. Many new measures have been proposed to improve the accuracy of similarity computation recently in some papers. But there are still various problems in those meas-

ures. The main contributions of this paper can be summarized as follows:

We propose a new similarity measure named RD similarity measure to calculate the similarity between users or items, which can achieve better similarity computation performances than other famous measures.

Based on the proposed RD similarity measure, we propose a new collaborative filtering algorithm (RDCF) with improved predictive accuracy and enhanced robustness, which can be applied to recommend various items to users in recommender systems.

We conduct a series of experiments based on three real datasets. According to comparisons with traditional measures and one of state-of-the-art measures, RDCF achieves better predictive performances.

The rest of this paper is organized as follows: Section 2 introduces related work. Section 3 presents our RD similarity measure. Section 4 presents our RDCF algorithm. Section 5 shows the experiments and Section 6 concludes the paper.

## 2. Related Work

CF algorithms are widely used in recommender systems. According to[3], CF algorithms can be grouped into two classes: memory-based CF and model-based CF. Memory-based CF approaches are usually classified into user-based approaches[6], item-based approaches [4-5] and combined approaches [12-13].

Similarity computation is the core step of CF. There are two similarity measures commonly adopted in CF recommender systems: Pearson's correlation coefficient (PCC)[3,9] and cosine similarity (COS)[4]. Because of the inherent problems of PCC and COS, researchers have proposed many improved similarity measures[14-16]. Mykhaylo et al.[7] claimed that COS and PCC gave incorrect predictive results, and pointed out that the inverse Euclidean distance was more suitable for calculating the similarity between rating vectors. Ma et al.[8] improved the PCC by adding a parameter to overcome the problem that the accuracy may be reduced.

Sun et al.[2] analyzed the drawbacks of PCC and COS .They hold that PCC cannot properly tackle the different rating styles between users, and that COS only considers the angle of two vectors but neglects their lengths. They proposed a Normal Recovery (NR) measure. They first normalized each row of the original user-item matrix by the lowest and the highest values of the same row, so that each row has a value range of [1], then they mapped the original user-item matrix into row-normal user-item matrix. X. Wu et al[1] analyzed the performance of NR and found that it had some problems as well. To obtain better results, they proposed another measure named ratio-based (RA) similarity measure. They regarded the ratio of two rating values of a specific item given by two users as the two users' consistency on this item, i.e., the single similarity, then they summed up all the single similarities together and get the average to be the final similarity between the two users. This similarity measure has been proved to have excellent predictive accuracy by their experiments. The authors believed it could overcome the limitations of PCC and COS. However, RA similarity measure is not without any problems. The RA cannot solve the same problem as PCC since it also neglects the different rating styles of users.

## 3. RD Similarity Measures

To overcome these extant shortcomings, we propose a new similarity measure named rating distance-based (RD) similarity measure in this paper.

Firstly, we take rating styles of users into consideration and define a formula named ratio of user rating standard deviation (RSD):

$$RSD(u,v) = \frac{\max(std(u), std(v))}{\min(std(u), std(v))} \quad (1)$$

where $\mathrm{std}(u)$ stands for the standard deviation of rating vector of user $u$.

In this formula, we take the standard deviation of user rating vector as an index to measure user rating diversities. Generally, smaller standard deviations means that the rating values of users are less reflective of their interest on the items they rated. Thus, this formula can be used to measure the degree of rating diversity difference between two users.

Usually, if two users give similar rating values on a specific item, we can infer that they have similar interest in this item relatively. Here we define the following formula to measure the degree of user rating difference on a single item, named single distance (SD):

$$SD(u,v,i) = \frac{|r_{u,i} - r_{v,i}|}{r_{\max} - r_{\min}} \quad (2)$$

where $r_{u,i}$ denotes the value of item $i$ rated by user $u$; $r_{\max}$ and $r_{\min}$ denote the highest and lowest values in the user-item matrix respectively. In most cases, $r_{\max}$ and $r_{\min}$ are exactly the highest and lowest values in the rating scale respectively. Take MovieLens-100K dataset as an example, whose rating values are in the interval of [1,5]. In this situation, $r_{\max}$ equals to 5 and $r_{\min}$ equals to 1. The calculation results of this formula are in the interval of [0,1], and a lower value reflects higher single similarity of user $u$ and $v$ on item $i$.

To get the single similarity of two users on a specific item, we need to know the difference of their interest in this item, which cannot be obtained by directly calculating the difference of rating values. Thus, we define a new distance computation standard combining RSD and SD to calculate user interest distance on a specific item, named composite distance standard (CDS):

$$CDS(u,v,i) = RSD(u,v) * SD(u,v,i) \quad (3)$$

Then the formula to calculate the single similarity of user $u$ and $v$ on item $i$ is defined as:

$$SS(u,v,i) = 1 - CDS(u,v,i) \quad (4)$$

The average of all the single similarities concerning both users $u$ and $v$ is exactly the similarity between the two users. So the new similarity measure RD is finally defined as the following formula:

$$Sim(u,v) = \frac{\sum_{i \in I} SS(u,v,i)}{|I|} = \frac{\sum_{i \in I}(1 - RDS(u,v) * SD(u,v,i))}{|I|}$$
$$= \frac{\sum_{i \in I}(1 - CDS(u,v,i))}{|I|} = \frac{\sum_{i \in I}(1 - \frac{\max(std(u), std(v))}{\min(std(u), std(v))} \frac{|r_{u,i} - r_{v,i}|}{r_{\max} - r_{\min}})}{|I|} \quad (5)$$
$$= 1 - \frac{\max(std(u), std(v))}{\min(std(u), std(v))} \frac{\sum_{i \in I} \frac{|r_{u,i} - r_{v,i}|}{r_{\max} - r_{\min}}}{|I|}$$

where $I = I_U \cap I_V$ denotes the set of items rated by both users $u$ and $v$; $|I|$ denotes the number of $I$; $r_{u,i}$ denotes the rating value of item $i$ rated by user $u$; $r_{\max}$ and $r_{\min}$ denote the highest and the lowest values in the user-item matrix, respectively; $std(u)$ denotes the standard deviation of the rating vector of user $u$.

Especially, the situation when users gives the same rating values for all the items they have rated makes $\min(std(u), std(v))$ equal to 0, which will make formula (5) fail to work. In this case, we can just ignore these abnormal users and regard the similarity computation result to be zero because their ratings cannot reflect their interest at all. When user $u$ gives very similar rating values for all the items he or she has rated, the value of $std(u)$ will be very low so that the similarity computation result may be a negative value. In this situation, these two users can be considered to be extremely dissimilar and the result is set to be zero. Thus, we optimize our RD similarity measure as follows:

$$sim(u,v) = \begin{cases} 0, & std(u) = 0 \text{ or } std(v) = 0 \\ \frac{1}{2}|1 - \frac{\max(std(u), std(v))}{\min(std(u), std(v))} \frac{\sum_{i \in I} \frac{|r_{u,i} - r_{v,i}|}{r_{\max} - r_{\min}}}{|I|}| + & \\ \frac{1}{2}(1 - \frac{\max(std(u), std(v))}{\min(std(u), std(v))} \frac{\sum_{i \in I} \frac{|r_{u,i} - r_{v,i}|}{r_{\max} - r_{\min}}}{|I|}), & otherwise \end{cases} \quad (6)$$

The results of this optimized formula are in the interval of [0,1], where 1 indicates the users are the same while 0 indicates they are not similar at all. And a higher result represents a higher similarity. When calculating the similarity between two items, we don't consider the difference of user rating styles. Thus, the formula of RD to measure the similarity between two items $i$ and $j$ can be expressed as follows:

$$Sim(i,j) = 1 - \frac{\sum_{u \in U} \frac{|r_{u,i} - r_{u,j}|}{r_{\max} - r_{\min}}}{|U|} \quad (7)$$

where $U = U_i \cap U_j$ is the set of the users who rated both items $i$ and $j$; $|U|$ is the number of $U$. The similarity values calculated by (7) are in the interval of [0,1], and a higher value represents a higher similarity.

## 4. RDCF

Based on our RD similarity measure, we propose an improved user-based collaborative filtering algorithm, named RDCF. We first generate a M*1 vector S, where the kth element is the standard deviation of all the rating values given by user k. Then S is normalized by the lowest and the highest values of S, so that the values of all the elements are in the interval of [0,1]. After that the original standard deviation vector S is mapped into normalized standard deviation vector Sn. And finally we obtain the normalized standard deviation of user $u$ by formula (8):

$$S_n(u) = \frac{std(u) - std_{\min}}{std_{\max} - std_{\min}} \quad (8)$$

Where $std_{max}$ and $std_{min}$ denote the maximum and minimum values in S, respectively. To predict the missing rating values, we defined our RDCF as follows:

$$\hat{r}_{u,i} = \begin{cases} \bar{r}_i, & S_n(u) \in [0, l) \\ \bar{r}_i + \frac{\sum_{u' \in U} Sim(u,u') \times r_{u',i}}{|U|}, & S_n(u) \in [l, 1] \end{cases} \quad (9)$$

where $\bar{r}_i$ denote the average rating value of item $i$ given by all users. $Sim(u,u')$ is calculated by RD similarity measure showed in formula (6).

$l$ is a parameter which we call similarity confidence threshold. The value of $l$ should be set in the interval of [0,1] and usually close to 0. The function of $l$ is to distinguish users with similar ratings from normal users. For users with similar ratings, it is unreasonable to recommend items based on the ratings of neighborhood. We recommend items based on their popularity which is calculated by $\bar{r}_i$.The criteria for judging which users are abnormal are different for different datasets. So the proper value of $l$ depends on the characteristics of datasets. By setting an appropriate value for $l$, RDCF can alleviate the interference of users with abnormal or malicious rating values. That will improve the predictive accuracy and enhance the robustness of recommender systems.

## 5. Experiments and Evaluations

In this section, we perform a series of experiments based on three real datasets to evaluate the performances of our RDCF and all the experimental results are obtained based on user-based CF.

### 5.1. Experimental datasets

Our experiments are conducted on three real datasets, ML-100K , ML-1M[23] and Film Trust[24], whose statistics are shown in Table 1. The density of Film Trust is 1.14 percent. Most users have rated less than 20 items and many of them have even rated less than 5 items, which means that the user-item matrix generated by using the data in Film Trust is very sparse. This dataset can help to validate the effectiveness of our algorithm in condition of data sparsity.

**Table 1. Statistics of three dataset**

| Dataset | Users | Items | Ratings | Density | Rating Scale |
|---|---|---|---|---|---|
| ML-100K | 943 | 1682 | 100000 | 6.30% | [1,5] |
| ML-1M | 6040 | 3706 | 1000209 | 4.47% | [1,5] |
| FilmTrust | 1508 | 2071 | 35497 | 1.14% | [0.5,4] |

### 5.2. Experimental setup

We use mean absolute error (MAE) to evaluate the predictive accuracy in our experiments. The MAE is the average absolute deviation of predictive values to actual values, which is defined as follow:

$$MAE = \frac{\sum_{m,i} \left| r_{m,i} - r_{m.i} \right|}{N} \qquad (10)$$

To perform a full evaluation and obtain more reliable results, five-fold cross validation is used in the experiments. Each dataset is evenly separated into five parts, in which four parts (80 percent of the dataset) are used as training set and one part (20 percent of the dataset) is used as test set.

### 5.3. Experimental results and analysis

#### 5.3.1. Impact of neighborhood size

The number of neighbors plays an important role in our RDCF algorithm, which determines how many similar users are employed to predict missing values. This experiment is set to explore the impact of neighborhood sizes on predictive accuracy, where neighborhood sizes range from 5 to 100 in increments of 5. As shown in Figure 1, MAE values begin to decline with the increase of K, and reach the bottom before they are going to rise. The predictive accuracy is not ideal if K is very small, which is likely to be caused by the similarity computation error. When using very few nearest neighbors, a single similarity computation mistake can cause great errors in predictive accuracy. Consider a special example where only one neighbor who has the highest similarity to the target user is involved in the prediction, if the similarity is miscalculated and they are actually not similar, we will obtain a completely wrong predictive result. The predictive results is not ideal as well if K is very big, which can be explained by the fact that the users ranking low in similarity among the large neighborhood is not actually similar to the target user.

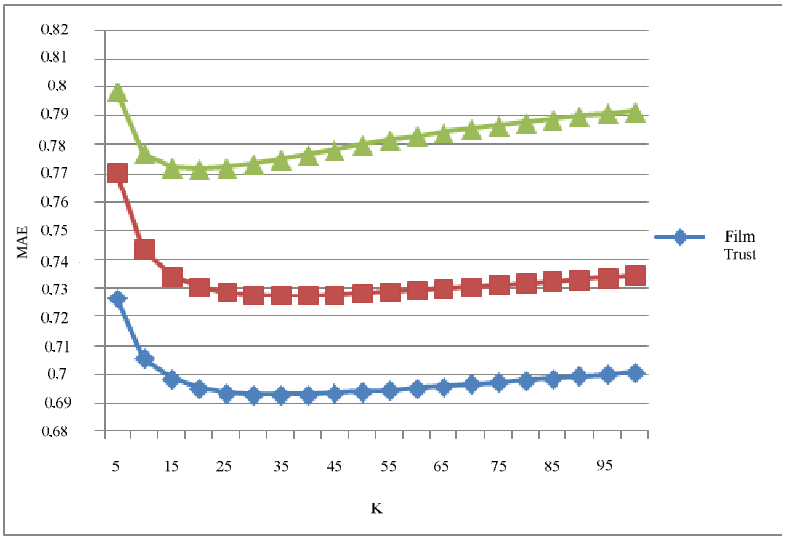


**Figure 1. Impact of neighborhood size K**

### 5.3.2. Impact of $l$

We set this experiment to explore the impact of similarity confidence threshold $l$ on predictive accuracy, where $l$ ranges from 0 to 0.3 in increments of 0.02. The experimental results showed in Figure 2. demonstrate that when $l$ is set to a specific value close to 0, RDCF achieves the best accuracy. And the optimum $l$ for Film Trust, ML-1M and ML-100K is 0.08, 0.06 and 0 respectively. When $l$ increases to a certain extent, the accuracy of RDCF will decline continuously with the increase of $l$ .

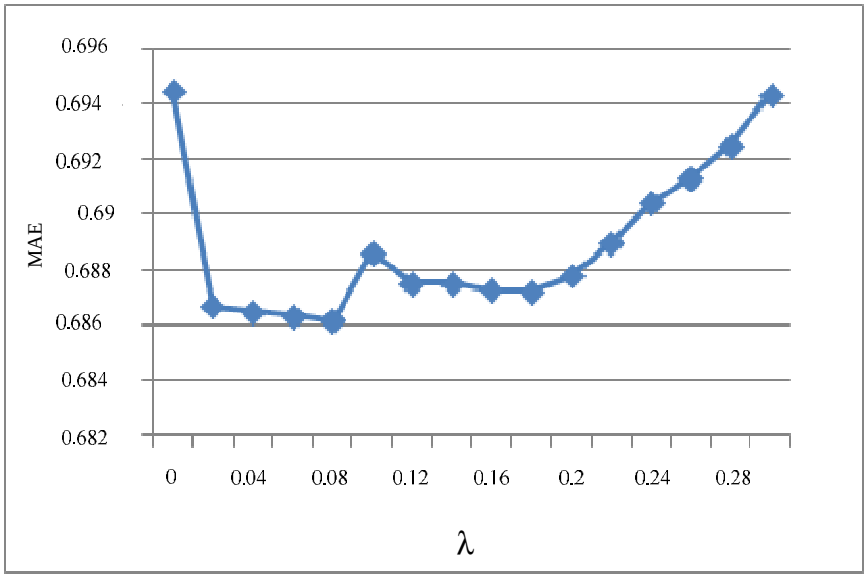


**Figure 2. Impact of $l$ , on datasets Film Trust (a)**

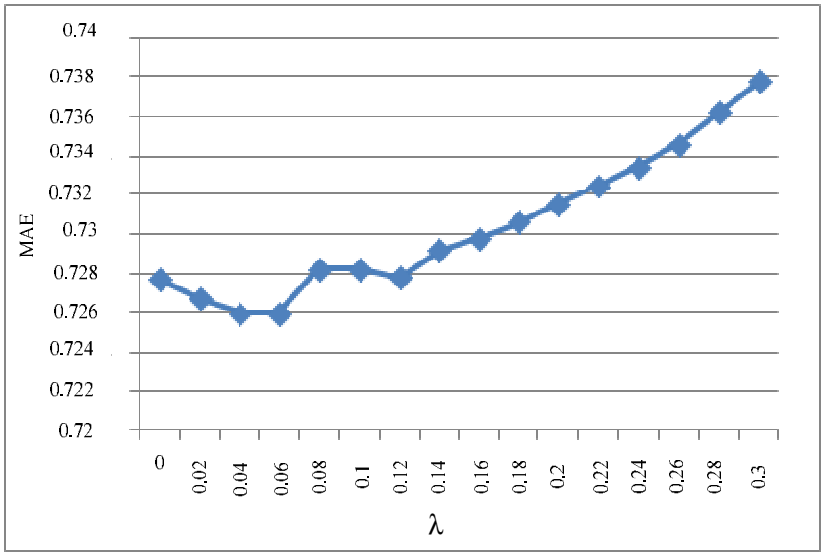


**Figure 2. Impact of $l$ , ML-1M(b)**

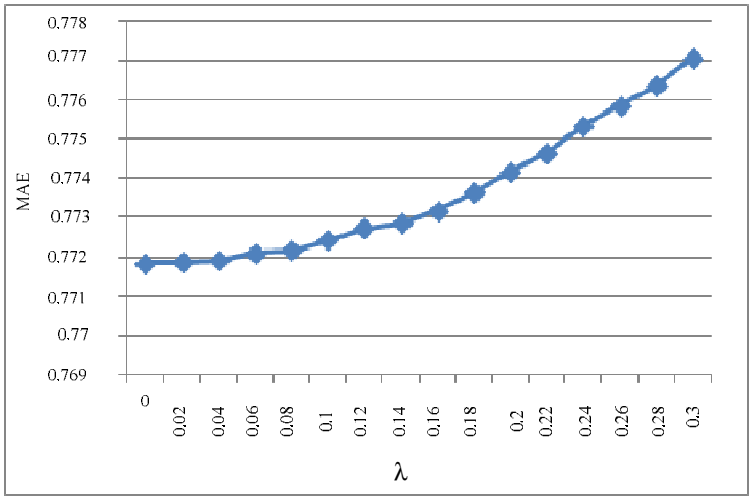


**Figure 2. Impact of $l$ , ML-100K(c)**

### 5.3.3. Performance comparisons of different measures

To show the effectiveness of our RDCF algorithm, we conduct this experiment to compare it with other three algorithms. The famous similarity measures selected by us for comparison are PCC, COS and RA. We use these three similarity measures combined with formula (14) to establish the corresponding CF algorithms. In RDCF algorithm, we employ $l$ =0.08, 0.06 and 0 for Film Trust, ML-1M and ML-100K respectively.

Compare MAE of the four algorithms in different neighborhood sizes

The purpose of this experiment is to compare MAE values of the four measures by varying neighborhood sizes from 5 to 50 with increments of 5. Experiment results shown in Figure 3. demonstrate that RDCF consistently achieves smaller MAE despite the change in neighborhood size K for all the three datasets. Compared with traditional measures COS and PCC, MAE of RDCF is much smaller, indicating that RDCF achieves better prediction accuracy to a large degree than those traditional measures. Also, prediction accuracy of RDCF is higher compared with RACF. Experimental results based on different datasets proved that RDCF can achieve better performances both in different data sizes and in condition of data sparsity.

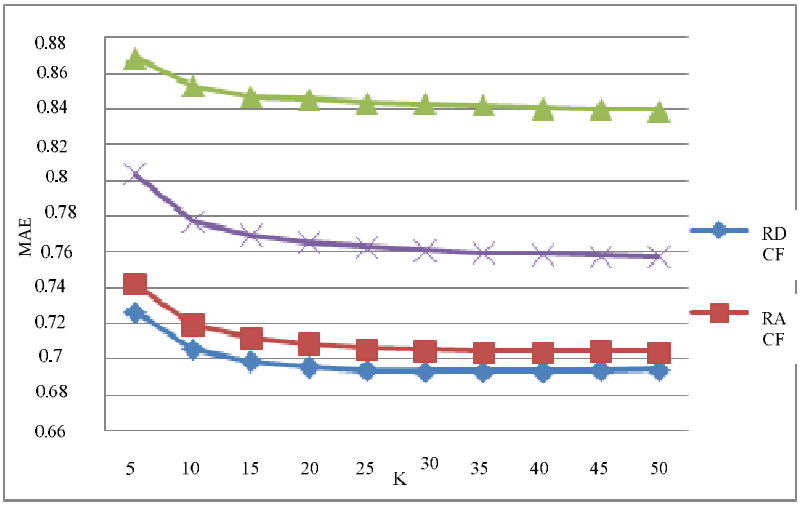


**Figure 3. MAE comparison in different neighborhood sizes, on datasets Film Trust (a)**

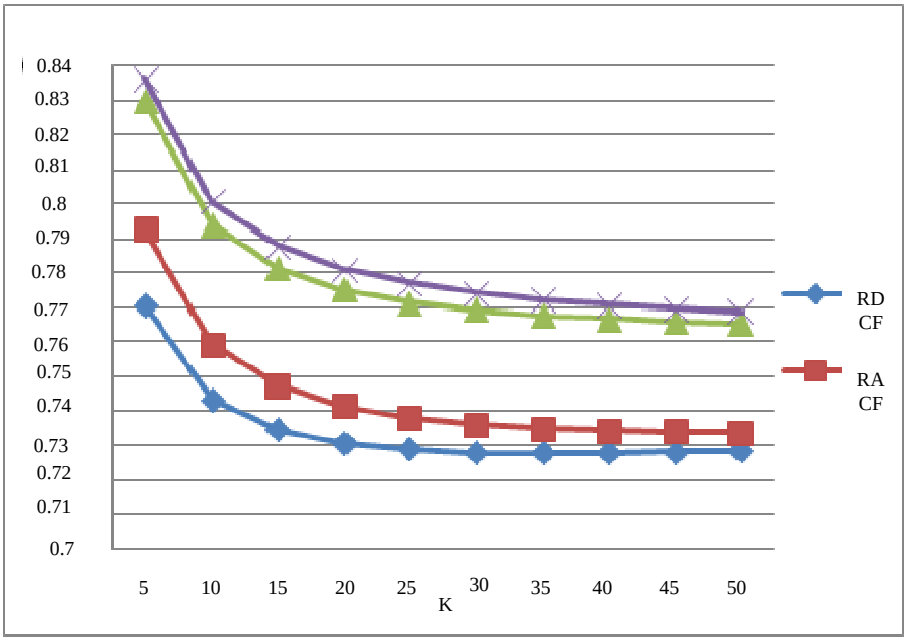


**Figure 3. MAE comparison in different neighborhood sizes, ML-1M(b)**

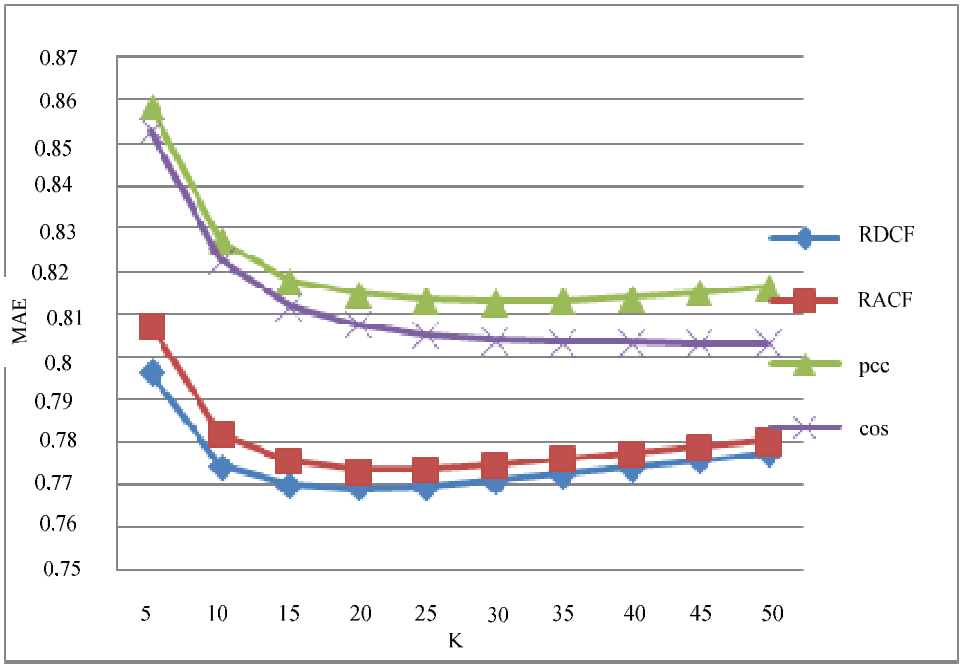


**Figure 3. MAE comparison in different neighborhood sizes, ML-100K(c)**

Compare MAE of the four algorithms with optimum parameters

Figure 4. show the MAE of the four algorithms with optimum neighborhood size K and similarity confidence threshold $l$ . For example in dataset ML-100K, we employ K=20, 20, 30 and 45 for RDCF, RACF, PCC, COS respectively, because those algorithms achieve their best performances when employing the corresponding K values. As shown in Fig.4, the best performance of RDCF outperforms that of other three algorithms. Especially, PCC is not suitable for predicting the missing values for dataset Film Trust because it will obtain much higher MAE than other three algorithms. In contrast, RDCF achieves much better predictive accuracy for Film Trust,

which reveals the superiority of RDCF in addressing data sparsity.

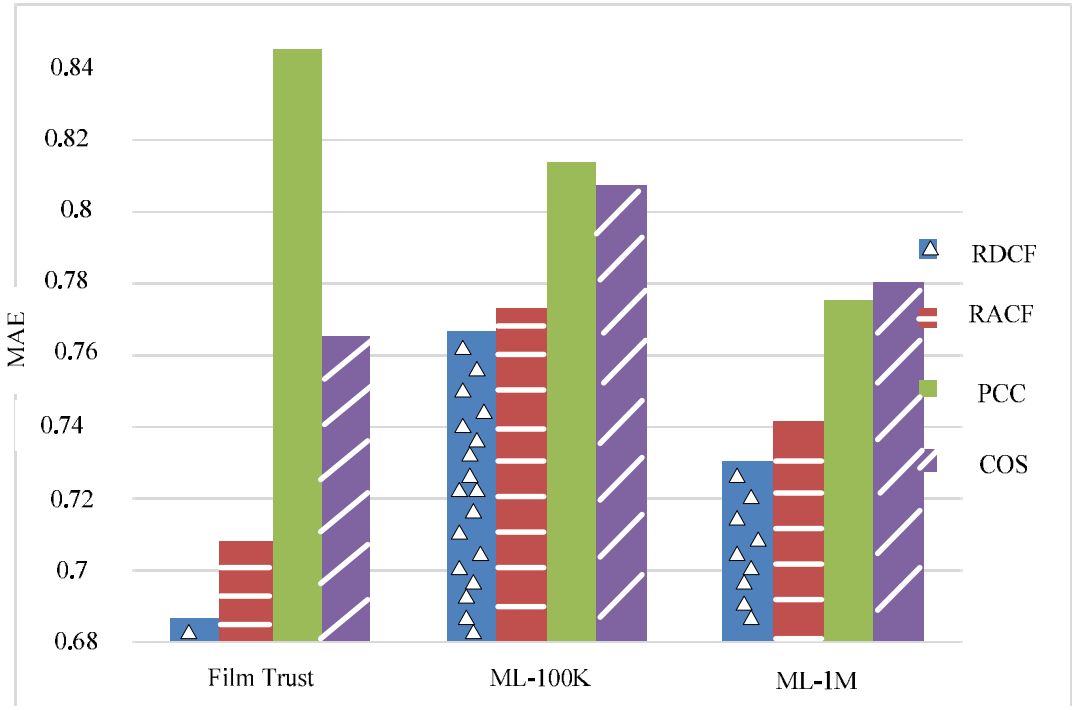


**Figure 4. MAE comparison with optimum parameters**

## 6. Conclusion

Similarity computation is a critical step in collaborative filtering recommender systems. In this paper, we propose a rating distance based (RD) similarity measure combined with statistical theories and a novel distance computation standard to address the thorny problems other measures cannot address properly. Based on RD similarity measure, we propose an improved collaborative filtering algorithm named RDCF. RDCF algorithm has enhanced robustness, which can alleviate the interference of users with abnormal or malicious ratings. We conduct a series of experiments based on three real datasets. According to comparisons with traditional measures and one of the most advanced measures, RDCF can calculate similarity more accurately and often achieves better predictive performances.

We plan to consider other statistical methods to perfect our algorithm in the future work. Furthermore, we intend to systematize our RDCF algorithm by utilizing the latest technology of machine learning and apply it to other application domains.

## References

[1] Wu X., Cheng B., Chen J. Collaborative filtering service recommendation based on a novel similarity computation method. IEEE Transactions on Services Computing. 2017, 10, 352 - 365.

[2] Sun H., Zheng Z., Chen J., Lyu M. Personalized web service recommendation via normal recovery collaborative filtering. IEEE Transactions on Services Computing. 2013, 6, 573–579.

[3] Breese J., Heckerman D. Empirical analysis of predictive algorithms for collaborative filtering. Proceedings of the 14th Conference Uncertainty in Artificial Intelligence. 1998, 43-52.

[4] Sarwar B., Karypis G., Konstan J., Riedl J. Item-based collaborative filtering recommendation algorithms. Proceedings of the 10th International World Wide Web Conference. 2001, 285-295.

[5] Deshpande M., Karypis G. Item-based top-n recommendation algorithms. ACM Transactions on Information Systems. 2004, 22, 143-177.

[6] He T.K., Chen Z.Y., Liu J., Zhou X.F., Du X.Z., Wang W.Q. An empirical study on user-topic rating based collaborative filtering methods. World Wide Web. 2017, 20, 815-829.

[7] Schwarz M., Lobur M., Stekh Y. Analysis of the effectiveness of similarity measures for recommender systems. Proceedings of the 14th International Conference the Experience of Designing and Application of CAD Systems in Microelectronics. 2017, 275–277.

[8] Ma H., King I., Lyu M. Effective missing data prediction for collaborative filtering. Proceedings of the 30th Annual International ACM SIGIR Conference. 2007, 39–46.

[9] Yang J., Li K. Recommendation based on rational inferences in collaborative filtering. Department of Electrical and Computer Engineering. 2008, 22, 105-114.

[10] Candillier L., Meyer F., Fessant F. Designing specific weighted similarity measures to improve collaborative filtering systems. Industrial Conference on Data Mining. 2008, 242–255.

[11] Tan Z., He L., Li H., Wang X. Rating personalization improves accuracy: a proportion-based baseline estimate model for

collaborative recommendation. International Conference on Collaborative Computing: Networking, Applications and Worksharing. 2016, 104–114.

[12] Jin R., Chai J., Si L. An automatic weighting scheme for collaborative filtering. Proceedings of the 27th Annual International ACM. 2004.

[13] Kant S., Mahara T. Merging user and item based collaborative filtering to alleviate data sparsity. International Journal of System Assurance Engineering and Management. 2018, 9, 173-179.

[14] Guo G., Zhang J., Yorke-Smith N. A novel Bayesian similarity measure for recommender systems. Proceedings of the IJCAI. 2013, 2619–2625.

[15] Guo G., Zhang J., YorkeSmith N. A novel evidence-based Bayesian similarity measure for recommender systems. ACM Transactions on the Web. 2016, 1-30.

[16] Ahn H.J. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. Information Science. 2008, 178, 37–51.

[17] Yang Z., Wu B., Zheng K., Wang X., Lei L. A survey of collaborative filtering-based recommender systems for mobile Internet applications. IEEE Access. 2016, 4, 3273–3287.

[18] Jorge A.M., Domingues M. Scalable online top-N recommender systems. International Conference on Electronic Commerce and Web Technologies. 2017, 278: 3–20.

[19] Burke R. Hybrid recommender systems: Survey and experiments. User Modeling and User-adapted Interaction. 2002, 12, 331–370.

[20] Lü L., Medo M., Yeung C.H., Zhang Y., Zhang Z., Zhou T. Recommender systems. Physics Reports. 2012, 519, 1–49.

[21] Wang L., Meng X., Zhang Y. Context-aware recommender systems. The Journal of Systems and Software. 2012, 23, 1–20.

[22] Ricci F., Rokach L., Shapira B. Context-aware recommender systems. Recommender Systems Handbook. 2010, 217–253.

[23] Harper F., Konstan J. The movielens datasets: history and context. ACM Transactions on Interactive Intelligent Systems. 2015.

[24] Golbeck J. Combining provenance with trust in social networks for semantic web content filtering. International Provenance and Annotation Workshop. 2006, 4145, 101-108.