



# A new user similarity model to improve the accuracy of collaborative filtering<sup>☆</sup>



Haifeng Liu<sup>\*</sup>, Zheng Hu, Ahmad Mian, Hui Tian, Xuzhen Zhu

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, China

## ARTICLE INFO

### Article history:

Received 19 February 2013

Received in revised form 3 November 2013

Accepted 7 November 2013

Available online 20 November 2013

### Keywords:

Collaborative filtering

User similarity

Recommended precision

Cold user

Recommender systems

## ABSTRACT

Collaborative filtering has become one of the most used approaches to provide personalized services for users. The key of this approach is to find similar users or items using user-item rating matrix so that the system can show recommendations for users. However, most approaches related to this approach are based on similarity algorithms, such as cosine, Pearson correlation coefficient, and mean squared difference. These methods are not much effective, especially in the cold user conditions. This paper presents a new user similarity model to improve the recommendation performance when only few ratings are available to calculate the similarities for each user. The model not only considers the local context information of user ratings, but also the global preference of user behavior. Experiments on three real data sets are implemented and compared with many state-of-the-art similarity measures. The results show the superiority of the new similarity model in recommended performance.

© 2013 The Authors. Published by Elsevier B.V. All rights reserved.

## 1. Introduction

Nowadays, more and more people have their own smart phone, tablet PC and other intelligent terminals. Which has enabled them to spend more time in accessing all kinds of social networks (such as Facebook and Twitter) and e-commerce sites (such as Amazon and eBay). However, the huge amount of available information and products makes them overwhelmed and indecisive. Users have to spend more time and energy in searching for their expected information. Even then, they cannot get satisfactory results. Fortunately, the behaviors of users can be tracked and recorded on the social networks and e-commerce sites. This makes it easier to analyze the preference of users. In this regard, recommender systems are used to recommend information of user expectations and provide personalized services through analyzing the user behaviors, such as the recommendation of photo groups in Flickr [1], the books in Amazon [2], videos in YouTube [3], and results in the Web search [4].

The collaborative filtering [5] has become the most widely used method to recommend items for users. It makes recommendation according to the similar users with the active user or the similar items with the items which are rated by the active user. The collaborative filtering includes memory-based method and model-based

method [6]. The memory-based method first calculates the similarities among users and then selects the most similar users as the neighbors of the active user. Finally, it gives the recommendations according to the neighbors. However, the model-based method first constructs a model to describe the behavior of users and, therefore, to predict the ratings of items. The memory-based method can give considerable recommended accuracy, but the computing time will grow rapidly with the increasing of users and items. In some conditions, it is difficult to respond in real-time. The model-based method tends to be faster in prediction time than the memory-based method, because the construction of the model can be finished in a considerable amount of time and this process is executed off-line. The shortcoming of the model-based method is that the recommendation performance is not as good for the memory-based method. In addition to collaborative filtering, content-based technique [7], social recommendation [8], semantic recommendation [9] are also applied in prediction of user preference.

This paper focuses on the recommended performance in memory-based collaborative filtering algorithms. The core of collaborative filtering is to calculate similarities among users or items. The generic traditional similarity measures, such as Pearson correlation coefficient [10], cosine [11], mean squared difference [6], are not enough to capture the effective similar users, especially for cold user who only rates a small number of items. This paper presents an improved heuristic similarity measure model. The new similarity model combines the local context for common ratings of each pair users and global preference of each user ratings. In order to test and verify the new similarity measure, experiments are

<sup>☆</sup> This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

<sup>\*</sup> Corresponding author. Tel.: +86 13436958586.

E-mail address: [haifengliu2011@gmail.com](mailto:haifengliu2011@gmail.com) (H. Liu).

implemented on three most used real data sets. In comparison with many state-of-the-art similarity measures, new model can show better recommended performance and better utilizes the ratings in cold user conditions.

## 2. Related work

Collaborative filtering (CF), as a kind of personalized recommendation technique, has been widely used in many domains [1–3,12,13]. However, collaborative filtering also suffers from a few of issues, for instance, cold start problem, data sparsity, scalability and so on. These problems seriously reduce the user experience. This paper focuses on how to improve the prediction accuracy. Collaborative filtering recommends items to users according to their preferences. Therefore, a history database of users' preference must be available. However, the database is always very sparse, that is, user only rates a small number of items. Up to now, there are many researchers who have focused on the prediction accuracy and proposed some solutions.

To improve the accuracy, many researchers have proposed some new similarity measures. Ahn [14] proposed a new similarity for collaborative filtering that is called PIP (Proximity-Impact-Popularity). This paper analyzed the disadvantages of Pearson correlation coefficient [10] and cosine similarity [11]. This new similarity considered three aspects: proximity, impact and popularity of the user ratings. But, this similarity considers only the local information of the ratings and does not consider the global preference of user ratings. Traditional Pearson correlation coefficient does not consider the size of the set of common users. To solve this problem, weighted Pearson correlation coefficient has been proposed [16]. It considers the idea of capturing the confidence which can be placed on the neighbor. The confidence will increase with the number of common rated items. Jamali and Ester [15] introduced a similarity measure based on the sigmoid function. This approach can weaken the similarity of small common items among users. The adjusted cosine similarity measure [14] was proposed to make up the shortage of traditional cosine similarity, however, it did not consider the preference of user ratings.

Bobadilla et al. [18] proposed a new metric which combined the Jaccard measure [17] and mean squared difference [6]. It assumed that these two measures could complement each other. Another new metric, which is called MJD (Mean-Jaccard-Difference), was proposed to solve the cold user problem. This metric includes three steps: first the selection of similarity measures, the new metric has six similarity measures after this step. Then, the weights of each similarity measure will be evaluated by neural network learning. Finally, the prediction can be obtained according to the new metric. Recently, a singularity based similarity measure (SM) [19] was also presented. This measure hypothesized that the results obtained by applying traditional similarity measures could be improved by taking contextual information. This paper first categorized the rating as positive and non-positive. Then, it computed the singularity values of each user and each item. It replaced the similarity with singularity value. The experiments verified the effectiveness of this approach. Moreover, Bobadilla et al. [20] introduced a significance based similarity measure. This measure first calculates three kinds of significances, which is the significance of an item, the significance of a user to recommend to other users and the significance of an item for a user. Then the traditional Pearson correlation coefficient or cosine similarity will be used to calculate the similarities among users according to the significance.

Data smoothing technique is another most used method to improve the recommend performance in collaborative filtering. Various sparsity measures [21] were used to enhance accuracy of collaborative filtering. These sparsity measures were computed

based on local and global similarities. Then, an estimating parameter scheme for weighting the various sparsity measures was proposed. The experimental results demonstrated that the proposed estimate parameter outperforms the schemes for which the parameter was kept constant on accuracy of prediction ratings. Ma et al. [22] proposed a partial missing data prediction algorithm, in which the information of both users and items was taken into account. In this algorithm, similarity threshold for users and items was set respectively and the missing data will be predicted if and only if, the intersection of the neighbors of user and the neighbors of item is not empty. The iterative prediction method [23] clusters the user and item respectively by using spectral clustering algorithm. Then, the iterative prediction technique is used to convert user-item sparse matrix to dense one based on the explicit ratings.

Beyond that, dimensionality reduction technique, such as principle component analysis (PCA) [24] and singular value decomposition (SVD) [25], is commonly used to alleviate the problem. Gong et al. [26] combined the SVD and item-based recommender in CF. It utilized the results of SVD to fill the missing ratings and then used the traditional item-based method to recommend. This combination method can increase the accuracy of system. Moreover, hybrid methods are also proposed. Szwabe et al. [27] investigated a hybrid recommendation method which was based on two-stage data processing—dealing with content features describing items and handling user behavioral data. This hybrid method combined random indexing (RI) technique and SVD to pre-process the content features. The experiments improved the recommendation accuracy without increasing the computational complexity. Probabilistic matrix factorization [28] is also combined in social recommendation to solve data sparsity.

Moreover, cluster-based smoothing method [29], support vector machine (SVM) [30], BP neural networks [31] and zero-sum reward and punishment mechanism [32] are also applied to smooth the missing ratings for the solution of accuracy in collaborative filtering.

## 3. The new similarity model

In this section, we first analyze the drawbacks of the existing similarity measures. Then, we introduce the motivation and hypothesis of the proposed similarity measure approach. Finally, we present the mathematic formalization of the proposed novel similarity measure approach. We assume that  $U = \{u_1, u_2, \dots, u_N\}$  and  $P = \{p_1, p_2, \dots, p_M\}$  are the set of users and items respectively. The user-item rating matrix is denoted as  $R = (r_{ij})_{N \times M}$ ,  $i = 1, 2, \dots, N; j = 1, 2, \dots, M$ .

### 3.1. The disadvantages of existing similarity measures

The Pearson correlation coefficient (PCC) and cosine (COS) similarity are the most widely used similarity measures in collaborative filtering. The formulas are defined as follows:

$$\text{sim}(u, v)^{\text{PCC}} = \frac{\sum_{p \in I} (r_{u,p} - \bar{r}_u)(r_{v,p} - \bar{r}_v)}{\sqrt{\sum_{p \in I} (r_{u,p} - \bar{r}_u)^2} \cdot \sqrt{\sum_{p \in I} (r_{v,p} - \bar{r}_v)^2}} \quad (1)$$

$$\text{sim}(u, v)^{\text{COS}} = \frac{\vec{r}_u \cdot \vec{r}_v}{\|\vec{r}_u\| \cdot \|\vec{r}_v\|} \quad (2)$$

where  $I$  represents the set of common rating items by user  $u$  and  $v$ .  $\bar{r}_u$  and  $\bar{r}_v$  is the average rating value of user  $u$  and  $v$  respectively.  $r_{u,p}$  and  $r_{v,p}$  denotes the rating of item  $p$  by user  $u$  and  $v$  respectively.  $\vec{r}_u$  and  $\vec{r}_v$  is the vector of the user  $u$  and  $v$  rated respectively. The magnitude of vector is represented as  $\|\cdot\|$ .

However, some shortages exist in both PCC and COS (described in follows). In order to overcome these drawbacks, many improved similarity measures have been introduced. Generally, the scale of ratings is absolute in recommender systems. The system can know which ratings are positive or negative. For considering the impact of positive and negative ratings, the constrained Pearson correlation coefficient (CPCC) [33] has been presented. The CPCC is defined as follows:

$$\text{sim}(u, v)^{\text{CPCC}} = \frac{\sum_{p \in I} (r_{u,p} - r_{\text{med}})(r_{v,p} - r_{\text{med}})}{\sqrt{\sum_{p \in I} (r_{u,p} - r_{\text{med}})^2} \cdot \sqrt{\sum_{p \in I} (r_{v,p} - r_{\text{med}})^2}} \quad (3)$$

where  $r_{\text{med}}$  is the median value in the rating scale. For example,  $r_{\text{med}}$  is 3 in a scale from 1 to 5 and, it is 4 in a scale from 1 to 7.

Intuitively, if both users have rated more common items, the similarity will be more credible. The weighted Pearson correlation coefficient (WPCC) [16] and sigmoid function based Pearson correlation coefficient (SPCC) [15] have also been proposed. The formulas of WPCC and SPCC can be defined as follows:

$$\text{sim}(u, v)^{\text{WPCC}} = \begin{cases} \text{sim}(u, v)^{\text{PCC}} \cdot \frac{|I|}{H}, & |I| \leq H \\ \text{sim}(u, v)^{\text{PCC}}, & \text{otherwise} \end{cases} \quad (4)$$

$$\text{sim}(u, v)^{\text{SPCC}} = \text{sim}(u, v)^{\text{PCC}} \cdot \frac{1}{1 + \exp\left(-\frac{|I|}{2}\right)} \quad (5)$$

where  $H$  is an experimental value and it is set 50 in [16]. Different people have different preferences of rating. Some people like to rate high, even they do not like the item very much. However, some people tend to rate low, even they like the items very much. The traditional cosine similarity does not account for the preference of the user's rating. For considering the preference of the user's rating, the adjusted cosine measure (ACOS) [14] has been introduced. The ACOS is defined as follows:

$$\text{sim}(u, v)^{\text{ACOS}} = \frac{\sum_{p \in P} (r_{u,p} - \bar{r}_u)(r_{v,p} - \bar{r}_v)}{\sqrt{\sum_{p \in P} (r_{u,p} - \bar{r}_u)^2} \cdot \sqrt{\sum_{p \in P} (r_{v,p} - \bar{r}_v)^2}} \quad (6)$$

where  $P$  is the set of all items. If user  $u$  has not rated the item  $p \in P$ , the rating  $r_{u,p}$  is zero.

Jaccard [17] and mean squared difference (MSD) [6] are another two widely used measures. The formulas are Eq. (7) and (8) respectively. Jaccard only considers the number of common ratings between two users. The basic idea is that users are more similar if they have more common ratings. The drawback is that it does not consider the absolute ratings. For example, user1 rates 5 and 4 on item1 and item2, user2 rates 1 and 2, user3 rates 4 and 5. Obviously, user1 and user 3 are more similar. MSD only considers the absolute ratings, but not consider the number of common ratings. The drawback is that it ignores the credibility of the similarity. Alike the previous example, assume that user1, user2 and user 3 have rated 5, 8 and 100 items respectively. Obviously, the similarity between user1 and user2 is more credible than the similarity between user1 and user3. Jaccard and MSD can be combined to form a new metric. The formula is as the Eq. (9).

$$\text{sim}(u, v)^{\text{Jaccard}} = \frac{|I_u \cap I_v|}{|I_u \cup I_v|} \quad (7)$$

$$\text{sim}(u, v)^{\text{MSD}} = 1 - \frac{\sum_{p \in I} (r_{u,p} - r_{v,p})^2}{|I|} \quad (8)$$

$$\text{sim}(u, v)^{\text{JMSD}} = \text{sim}(u, v)^{\text{Jaccard}} \cdot \text{sim}(u, v)^{\text{MSD}} \quad (9)$$

where  $r_u$  and  $r_v$  represents the set of items by user  $u$  and  $v$  rated respectively.

**Table 1**

An example of the user-item rating matrix. The missing ratings are represented by the symbol  $-$ .

	Item1	Item2	Item3	Item4
User1	4	3	5	4
User2	5	3	-	-
User3	4	3	3	4
User4	2	1	-	-
User5	4	2	-	-

Although, many similarity measures have been proposed and they make up some drawbacks of the traditional similarity methods. These similarity measures still have some drawbacks. In this section, we will analyze these similarity methods in detail and show the shortages of these methods.

Table 1 gives an example of a user-item rating matrix. We assume that there are four items and five users in the systems. The missing ratings of the rating matrix are represented by the symbol. Then we calculate the similarities of users in the table according to those similarity measures described the above. Fig. 1 shows the results of the user similarities in Table 1. Since user similarity matrix is symmetric, we only show partial values.

The main drawbacks are described as follows:

- (1) Low similarity regardless of the similar ratings by two users. Fig. 1(a) gives the user similarity matrix according to Pearson correlation coefficient. From Table 1 we can see that the User1 and User3 have very similar ratings. The rating vector is (4, 3, 5, 4) and (4, 3, 3, 4) for User1 and User3 respectively. However, we can see that the similarity of these two users is zero in Fig. 1(a). This drawback is still in the SPCC (Fig. 1(c)). And the CPCC has certain improvement, which the similarity is 0.577 (Fig. 1(b)). The ACOS also has this problem. In Fig. 1(e), the similarity between User1 and User3 is zero as well.
- (2) High similarity regardless of the difference between the two user's ratings. Fig. 1(a) also shows that the user can obtain high correlation regardless of the difference in the ratings of both users. For example, the rating vectors of the both User2 and User4 are (5, 3, -, -) and (2, 1, -, -). However, the similarity is 1.0 between these two users. In Fig. 1(c), the SPCC similarity of these two users is 0.731. This is also very high. If two vectors are on the same line, the similarity will be set 1 according to cosine regardless of the difference between both users. This can be seen in Fig. 1(d). The two vectors of User 4 and User5 are (2, 1, -, -) and (4, 2, -, -) respectively. The similarity between them is 1 according to cosine. This shortage can be solved by adjusted cosine measure (ACOS). We can see that the similarity becomes 0.1 between User4 and User5 according to ACOS in Fig. 1(e).
- (3) Ignoring the proportion of common ratings will lead low accuracy.

Mean squared difference (MSD) only calculates the average difference between both users, but ignores the proportion of common ratings. This may lead to the low accuracy. In Fig. 1(f), the similarity between User1 and User2 is 0.98 and, the similarity between User1 and User3 is 0.96. But, in fact, from Table 1 we can see that the similarity between User1 and User3 should be higher than the similarity between User1 and User2. This is because the MSD does not consider the proportion of common ratings. In Table 1, the proportion of common ratings between User1 and User2 is 1/3 (the calculation of the proportion is the number of common ratings divided by the total number of User1 and User2 ratings). However, the proportion is 1/2.

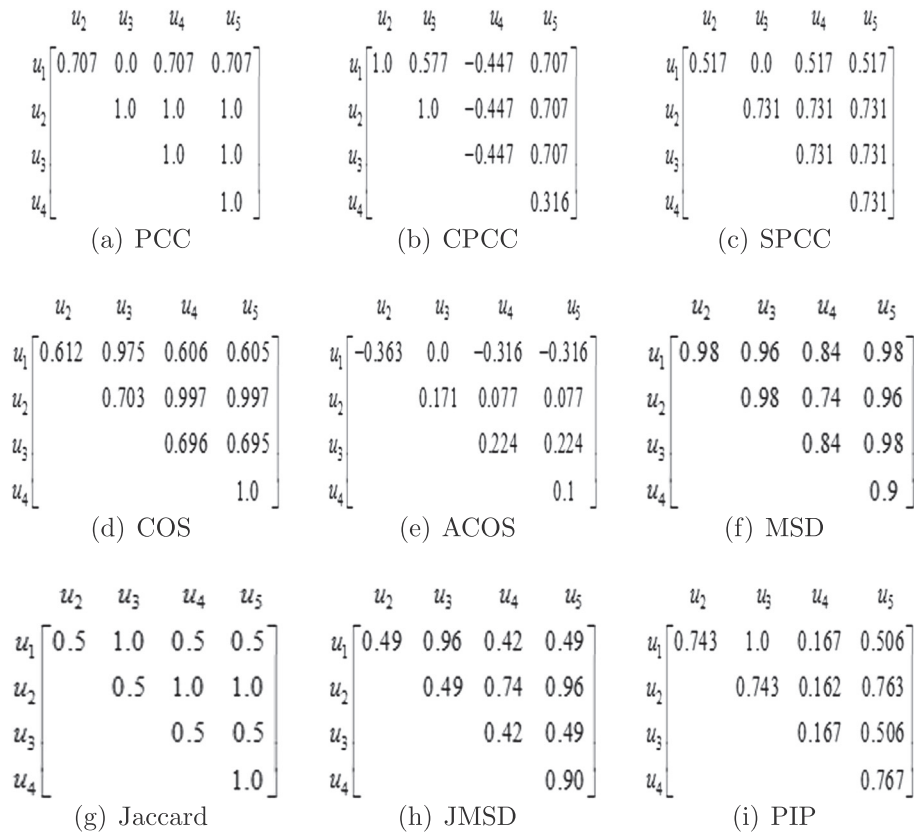


Fig. 1. The user similarity matrix in Table 1, according to all kinds of similarity measures.

- (4) Discarding the absolute value of rating will become difficult to distinguish different users.

On the contrary, the Jaccard similarity only considers the proportion of common rating, and does not consider the absolute value of rating. This leads to the difficulty of distinguishing between the users. In Fig. 1(g), we note that there are only two kinds of similarities: 1.0 and 0.5. The similarities between the users who have rated four items and the users who have rated two items are 0.5. For example, the similarity between User1 and User2. However, the similarities between the both users who have rated two items are 1.0. For instance, the similarity between User2 and User5. Moreover, we also notice that the similarity between User2 and User5 is the same as the similarity between User2 and User 4. But, obviously, from Table 1 we can see that the former should be higher than the latter.

- (5) The combination of Jaccard and MSD only solves partial problems.

The combination of Jaccard and MSD can make up for the partial shortages of Jaccard and MSD. In Fig. 1(g), we notice that the similarity becomes diversity. Different users will have different similarities. But it is not thorough, for example, the similarity between User1 and User4 is also the same as the similarity between User3 and User4. The low accuracy has certain improvement which exists in Fig. 1(f). For instance, the similarity between User1 and User3 is higher than the similarity between User1 and User2.

### 3.2. The motivations of the new similarity measure model

In previous section, we have analyzed the drawbacks of the traditional similarity measures and the improved variants. In most recommender systems, most users only rate a small number of

items. This leads to very low accurate similarity based on the previous similarity measures. In order to improve the recommended accuracy, this paper proposes an improved heuristic similarity measure model. First, we introduce the initial heuristic measure. Then we analyze its shortages and present the motivations of the improved novel heuristic similarity measure.

#### 3.2.1. The initial heuristic similarity measure

This heuristic similarity measure is composed of three factors of similarity, *Proximity*, *Impact* and *Popularity*. And hence, the measure is named *PIP*.

Fig. 2 shows the basic idea of the PIP similarity measure. The first factor, *Proximity* factor, not only calculates the absolute difference between two ratings, but also considers whether these ratings are in agreement or not, giving penalty to ratings in disagreement. The *Impact* factor represents how strongly an item is preferred or disliked by users. If two users have rated 5 on an item, it will show more strong prefer than they rate 4. We note that it is penalized repeatedly on the computation of *Proximity* and *Impact*, when two ratings are not in agreement. This is not very reasonable. The last factor is *Popularity*. It denotes how common two user's ratings have. If both users average rating has a large difference with the average of total users' ratings, the two ratings can provide more information about the similarity of the two users.

The PIP similarity between user  $u$  and  $v$  can be calculated as:

$$\text{sim}(u, v)^{\text{PIP}} = \sum_{p \in I} \text{PIP}(r_{u,p}, r_{v,p}) \quad (10)$$

where  $\text{PIP}(r_{u,p}, r_{v,p})$  is the PIP value for the two ratings  $r_{u,p}$  and  $r_{v,p}$  on item  $p \in I$  by user  $u$  and  $v$  respectively. It can be defined as follows:

$$\text{PIP}(r_{u,p}, r_{v,p}) = \text{Proximity}(r_{u,p}, r_{v,p}) \cdot \text{Impact}(r_{u,p}, r_{v,p}) \cdot \text{Popularity}(r_{u,p}, r_{v,p}) \quad (11)$$



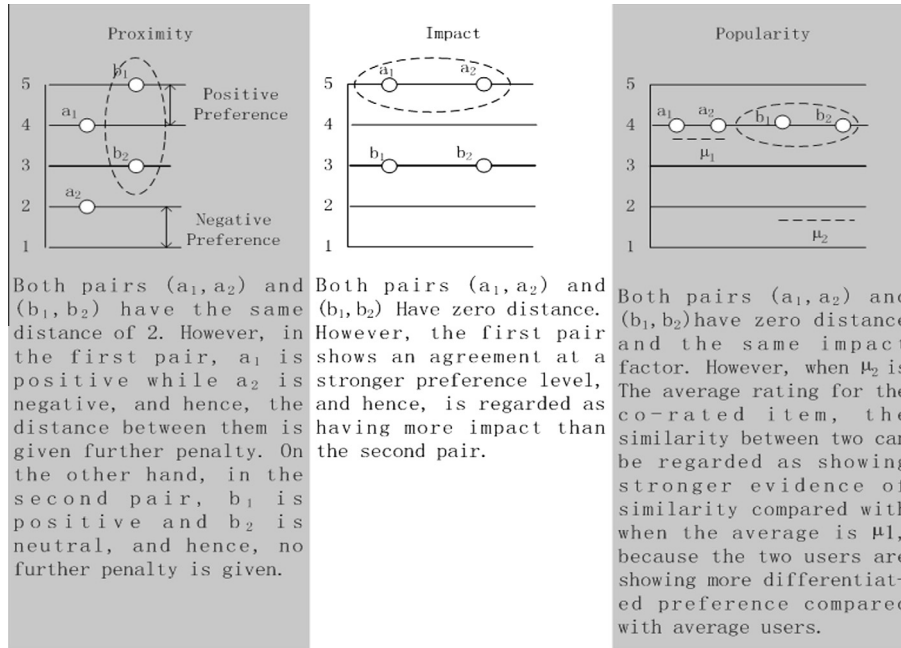


Fig. 2. The description of the basic idea of the PIP (the figure is from [14]).

The detailed calculation about the three factors can be found in [14].

### 3.2.2. The motivations

From the description of the previous section, we notice that the PIP similarity measure only considers the absolute value of the rating and penalize repeatedly on the first two factors. However, the analysis in Section 3.1 shows that it will be sometimes misleading, where the similarity between similar users may be lower than the similarity between dissimilar users. In Fig. 1 (i), it shows the user similarity matrix of the Table 1. We notice that the PIP similarity between User3 and User5 is lower than the PIP similarity between User4 and User5. However, from Table 1 we can see that the former is more similar than the latter.

The motivations for our improved heuristic similarity measure approach are as follows:

- (1) The similarity measure not only considers the absolute ratings, but also considers the proportion of the common ratings. The initial PIP measure only considers the set of common ratings and the absolute value, but not considers the proportion of the common ratings. This will lead to low accuracy. For example, user1 and user2 have four common rated items, where user1 and user2 have rated 6 and 8 items respectively. It will be more similar than user1 and user3 who have four common rated items, where user3 has rated 100 items. Hence, we adopt the idea of Jaccard measure to improve the PIP measure for considering the proportion of common ratings.
- (2) The similarity not only is decided by local context, but also the global preference of user behavior.

We notice that the initial PIP measure only considers the local context information of common ratings. We have analyzed that the misleading of similarity still exists in the PIP. Just like the MSD, this misleading does not eliminate by combining only the Jaccard measure. This can be seen in Fig. 1(h). There exists the misleading of similarity in the JMSD also, although it considers the proportion of common ratings. The reason of the misleading of the similarity is

because these similarities only consider the local context information of the ratings. Hence, in order to eliminate the misleading, we consider the global information about the preference of the user behavior.

- (3) The similarity measure should be normalized and easily combined with other similarity measures.

From the paper [14], we see that the initial formula of PIP similarity is very complex. It uses different formulas in different conditions. Moreover, the initial PIP similarity is not normalized. It is difficult to combine with other similarity measures. The most important is that the calculation of the PIP similarity is linear mainly. A good similarity measure should amplify the positive factors and restrain the negative factors. Hence, we consider building the model of the calculation of similarity measure with a non-linear formula.

### 3.3. Formalization of the new similarity measure model

#### 3.3.1. The formalization

In this section, we give the mathematic formalization of the proposed novel similarity measure approach. The previous section describes that the initial PIP similarity formula is too complex and not normalized. In order to punish the bad similarity and reward the good similarity, we adopt a non-linear function in our model. That is sigmoid function. Moreover, we will call the improved PIP measure as PSS (Proximity-Significance-Singularity). The user PSS similarity can be calculated as follows:

$$\text{sim}(u, v)^{\text{PSS}} = \sum_{p \in I} \text{PSS}(r_{u,p}, r_{v,p}) \quad (12)$$

where the  $\text{PSS}(r_{u,p}, r_{v,p})$  is the PSS value of user  $u$  and  $v$ , it is defined as follows:

$$\text{PSS}(r_{u,p}, r_{v,p}) = \text{Proximity}(r_{u,p}, r_{v,p}) \times \text{Significance}(r_{u,p}, r_{v,p}) \times \text{Singularity}(r_{u,p}, r_{v,p}) \quad (13)$$

We can see that the PSS measure is also composed of three factors of similarity, *Proximity*, *Significance* and *Singularity*. *Proximity* is

the similar as the *PIP*. However, it only considers the distance between two ratings. The second factor is *Significance*. We assume that the ratings are more significance if two ratings are more distant from the median rating. For example, if two user rate two items as (4,4) or (2,2). We think it is more significant than two user give (5,3) or (4,2). The third factor is called *Singularity*. This factor represents how two ratings are different with other ratings. The formalizations of the three factors are defined as follows:

$$\begin{aligned} \text{Proximity}(r_{u,p}, r_{v,p}) &= 1 - \frac{1}{1 + \exp(-|r_{u,p} - r_{v,p}|)} \\ \text{Significance}(r_{u,p}, r_{v,p}) &= \frac{1}{1 + \exp(-|r_{u,p} - r_{med}| \cdot |r_{v,p} - r_{med}|)} \\ \text{Singularity}(r_{u,p}, r_{v,p}) &= 1 - \frac{1}{1 + \exp(-\left|\frac{r_{u,p} + r_{v,p}}{2} - \mu_p\right|)} \end{aligned} \quad (14)$$

where  $\mu_p$  is the average rating of item  $p$ .  $r_{u,p}$  is the rating of item  $p$  by user  $u$ . Unlike the three factors of the initial *PIP*, each factor belongs to (0, 1) in our model.

In Section 3.2.2, we analyze that the proportion of common ratings is a very important factor. In our model, we also consider this factor and it is different with the Eq. (7). We modify the formula to punish the small proportion of common ratings. It is defined as follows:

$$\text{sim}(u, v)^{\text{Jaccard}} = \frac{|I_u \cap I_v|}{|I_u| \times |I_v|} \quad (15)$$

We can combine PSS with the modified Jaccard as a new similarity measure. That is called *JPSS*. The formalization is as follow:

$$\text{sim}(u, v)^{\text{JPSS}} = \text{sim}(u, v)^{\text{PSS}} \cdot \text{sim}(u, v)^{\text{Jaccard}} \quad (16)$$

Further, we should consider the preference of each user. Different users have different rating preferences. Some users prefer giving high ratings. Some users tend to rate low value. In order to reflect this behavior preference, we adopt the mean and variance of the rating to model the user preference. The user rating preference based on similarity measure can be defined as follows:

$$\text{sim}(u, v)^{\text{URP}} = 1 - \frac{1}{1 + \exp(-|\mu_u - \mu_v| \cdot |\sigma_u - \sigma_v|)} \quad (17)$$

where  $\mu_u$  and  $\mu_v$  is the mean rating of user  $u$  and  $v$  respectively. And  $\mu_u = \sum_{p \in I_u} r_{u,p} / |I_u|$ . The  $\sigma_u$  and  $\sigma_v$  represents the standard variance of user  $u$  and  $v$ . The calculation is  $\sigma_u = \sqrt{\sum_{p \in I_u} (r_{u,p} - \bar{r}_u)^2 / |I_u|}$ .

We can obtain the final formalization by combining the formula (16) and (17), which we called improved new heuristic similarity model (NHSM). The formalization is as follows:

$$\text{sim}(u, v)^{\text{NHSM}} = \text{sim}(u, v)^{\text{JPSS}} \cdot \text{sim}(u, v)^{\text{URP}} \quad (18)$$

We note that the NHSM similarity is in (0, 1), because each part is from 0 to 1.

### 3.3.2. Discussions on the new heuristic similarity model

We discuss the drawbacks of the existing similarity measures in Section 3.1. We now show that the improved new heuristic similarity model (NHSM) can successfully overcome these drawbacks. Fig. 3 shows the user similarity matrix of Table 1 according the improved model (NHSM).

First, from the Fig. 3 we can see that the similarity between User1 and User3 is higher than the similarity between User1 and User2. However, this is not accurate in PCC, CPCC, SPCC, ACOS and MSD. This indicates that the NHSM similarity measure can overcome the drawback of low similarity regardless of similar ratings by two users.

	$u_2$	$u_3$	$u_4$	$u_5$
$u_1$	0.02089	0.05520	0.00475	0.02440
$u_2$		0.0183	0.00464	0.03561
$u_3$			0.00636	0.02500
$u_4$				0.01531

Fig. 3. The user similarity matrix of Table 1 according to the proposed new heuristic similarity model (NHSM). The NHSM similarity measure overcomes the drawbacks effectively described in Section 3.1.

Second, the similarity between User3 and User5 is also higher than the similarity between User4 and User5. However, the misleading still exists in COS, ACOS, Jaccard, JMSD and *PIP* similarity. This demonstrates that the NHSM similarity measure can avoid the misleading.

Third, if high difference exists between the two user's ratings, the similarity will be very small. For example, the similarity between User2 and User4 is very low, it is about 0.00464. It is the least similarity in the user similarity matrix. From Table 1, we can see that the rating vectors of these two users are (5, 3, −, −) and (2, 1, −, −) respectively. It has very low similarity indeed. Even when the difference is very small, the NHSM also can distinguish the similarities accurately. For instance, the similarity between User1 and User5 is slightly higher than the similarity between User1 and User2.

Fourth, each user becomes comparable, that is each user has different similarities. This can be seen in Fig. 3. Each pair has different similarities. However, this is not the case in existing similarity measures. This also can be seen from Fig. 1.

## 4. Experiments

### 4.1. Data set

The two data sets of MovieLens (<http://www.movie-lens.umn.edu>) and Epinions are used in our experiments. The first in MovieLens is called ML-100K, there are 100,000 ratings with 943 persons and 1682 movies. Another is ML-1M, it includes 6040 users and 3952 movies with 1,000,209 ratings. In both data sets, each person has rated at least 20 movies. The user profile includes age, sex, and profession. The movie includes 19 types. The density of the user-item matrix is 6.3% in ML-100K and 4.1% in ML-1M.

Epinions (<http://www.epinions.com/>) data set is collected from epinions.com. Epinions founded in 1999 is a product and shop review site where users can review items (such as movies, books, and software) and users can also assign items numeric ratings in the range 1–5. Moreover, users can express their trust to other users, i.e. reviewers whose reviews and ratings are helpful and valuable to me. The Epinions data set consists of 49,289 users who have rated a total of 139,738 different items at least once. There are 40,163 users who have rated at least one item. The total number of reviews is 664,824. The sparseness of the data set is hence more than 99.99%.

We choose these three data sets, is because they are the most used data sets by researchers and developers in collaborative filtering domain.

For demonstrating the performance of the improved similarity model, each data set is divided into two parts, 20% of all persons are selected to be testing users, and the remaining as training users. In MovieLens, we only choose one to ten items as the training ratings from each testing user and others as the testing ratings. Since, Epinions is too sparse, we select 20% items for each testing user as the training ratings, others as the testing ratings. Just like the most researchers, for measuring the prediction accuracy, we

conducted a k-fold cross validation by randomly choosing different training and testing sets.

#### 4.2. Evaluation metrics

Many researchers predict the rating of item by the user. The most used metrics are Mean Absolute Error (MAE) [14,19] and Rooted Mean Squared Error (RMSE) [21]. However, in many cases best MAE or RMSE is not equal to the best user satisfaction. The precision and recall are better in top-N recommendation [34].

Hence, in order to estimate the performance of the proposed similarity model, the prediction accuracy is measured with two popular used metrics.

**Recall.** The recall score is the average proportion of items from testing set that appear among of the ranked list from the training set [35]. This measure should be as high as possible for good performance. Assuming  $M_T$  is the number of items which are in the testing set and liked by the active user,  $n$  is the amount of items which the testing user likes and appears in the recommended list. Hence, the recall is computed as follows:

$$\text{Recall} = \frac{n}{M_T} \quad (19)$$

**Precision.** The precision is the proportion of recommended items that the testing users actually liked in the testing set [36]. This measure is also as high as possible for good performance. The precision is computed as follows:

$$\text{Precision} = \frac{n}{\text{TopN}} \quad (20)$$

#### 4.3. Compared methods

In our experiments, we compare our improved similarity with many state-of-the-art similarity measures described in Section 1.

Except the initial PIP similarity measure [14], we also compare our similarity with some new similarity metrics. SM (Singularity Measure) [19] is based on the singularity of item. The definition is as follows:

$$\begin{aligned} \text{sim}(u, v)^{\text{SM}} &= \frac{1}{3} \cdot (\text{Sing}(A) + \text{Sing}(B) + \text{Sing}(C)) \\ \text{Sing}(A) &= \frac{1}{|A|} \sum_{i \in A} [1 - (r_{u,i} - r_{v,i})^2] (s_p^i)^2 \\ \text{Sing}(B) &= \frac{1}{|B|} \sum_{i \in B} [1 - (r_{u,i} - r_{v,i})^2] (s_n^i)^2 \\ \text{Sing}(C) &= \frac{1}{|C|} \sum_{i \in C} [1 - (r_{u,i} - r_{v,i})^2] (s_p^i \cdot s_n^i) \end{aligned} \quad (21)$$

where  $A$  and  $B$  is the set, in which the ratings of both user  $u$  and  $v$  are positive and negative respectively. In set  $C$ , one user's rating is positive and another is negative.  $s_p^i$  is the positive singularity of item  $i$ .  $s_n^i$  is the negative singularity of item  $i$ .

MJD (Mean-Jaccard-Difference) [18] is a combination method of several similarity metrics. It only considers the similarity measures which cause a very worse recommendation quality results. The final formula is defined as follow:

$$\text{sim}(u, v)^{\text{MJD}} = \frac{1}{6} \left( w_1 \cdot d_{uv}^0 + w_2 \cdot d_{uv}^1 + w_3 \cdot d_{uv}^3 + w_4 \cdot d_{uv}^4 + w_5 \cdot \mu_{uv} + w_6 \cdot \text{Jaccard}_{uv} \right) \quad (22)$$

where  $d_{uv}^k$  represents the set of item, which the difference of ratings by user  $u$  and  $v$  is equal to  $k$ .  $\mu_{uv}$  is the mean squared difference and  $\text{Jaccard}_{uv}$  is the Jaccard similarity between user  $u$  and  $v$ .  $w_i$  is the weight of the  $i$  basic similarity metric, which can be obtained based on the neural network learning.

#### 4.4. Performance comparison

In this section, several experiments are conducted on the two data sets and we compare the improved similarity measure with many other measures. In collaborative filtering, there are two parameters which can impact the performance of recommendation, that is, the number of nearest neighbors and the number of recommendations. We will compare the results with different values of these two parameters. Moreover, the recommended lists can be obtained by two steps. First, we predict the ratings for the user unrated items according to his/her nearest neighbors. Second, the *TopN* items with highest predicted values will be selected as the recommended lists.

##### 4.4.1. The effect of the number of nearest neighbor

Assume  $K$  that denotes the number of nearest neighbor. Different  $K$  will lead different recommendation accuracies. We first analyze the impact of the  $K$  to the performance.

Fig. 4 gives the performances of different similarity measures with different number of nearest neighbor on the ML-100K data set. In Fig. 4(a), the recalls of all the similarity measures decrease with the increasing of the number of nearest neighbor. We note that the improved similarity measure (NHSM) obtains the better recall than most other similarities when the  $K$  is not very large. However, it is worse than cosine similarity.

Compared with the PIP similarity measure, our improved similarity measure has remarkable improvement. The recall of PIP is better than the Pearson when the  $K$  is more than 40. However, the PIP is worse than the adjusted cosine, cosine, MJD and sigmoid-based similarity measures. The most improvement of recall of NHSM can be reached more than two times compared with the PIP.

In Fig. 4(a), the Pearson measure has the worst recall when the  $K$  is more than 50. The variants of Pearson will have a better recall when the  $K$  is less than 50, especially the sigmoid-based Pearson measure can surpass the Pearson in the whole range. The variant of cosine is also better than cosine when  $K$  is larger than 70.

In Fig. 4(b), the precision of the improved similarity measure can obtain the best result when the  $K$  is less than 10 and the result is worse than cosine similarity when the  $K$  is more than 20. The precision of PIP similarity is more stable than NHSM on the whole range. Compared with the PIP, our improved similarity measure can improve 100% when the  $K$  is 10. Except the NHSM, the COS has the best precision. Compared with the ACOS, the NHSM still improves 30.8% when the  $K$  gets 20. Pearson similarity has the worse precision when the  $K$  is more than 50.

Moreover, we notice that the recall and precision obtained by some measures (such as PIP) increase and others (such as ACOS) decrease as  $K$  increases. That is because some measures are sensitive to false neighbors. False neighbor means that the similarity between two users is high, but in fact, their preferences are not similar, this often leads by the data sparsity. Hence, although our measure can provide better recall and precision, it is not stable enough.

Fig. 5 shows the change of performance of different similarity measures with different  $K$ -neighbors on the ML-1M data set. From the figure we see that not only the recall but also the precision of all the similarity measures decrease with the increasing of the  $K$ . However, the NHSM, COS and MJD decrease faster than others.

In Fig. 5(a), our improved similarity measure (NHSM) can obtain the best recall when the  $K$  is less than 20. The recall of NHSM is worse than the COS and MJD when the number of neighbor is more than 20. The recall of PIP is very bad but it is better than the WPCC, SPCC and SM. The NHSM has about 166% improvement compared with the PIP when the  $K$  gets 10. Except the NHSM, the COS and MJD can surpass others when the  $K$  is more than 20 and the ACOS

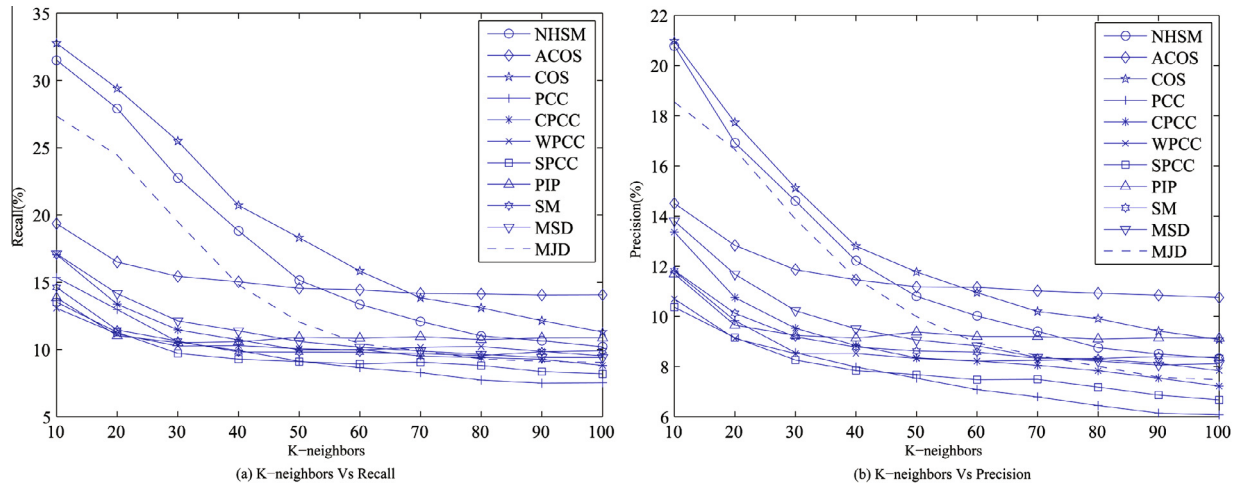


Fig. 4. The performances of different similarity measures with different K-neighbors on ML-100K data set.

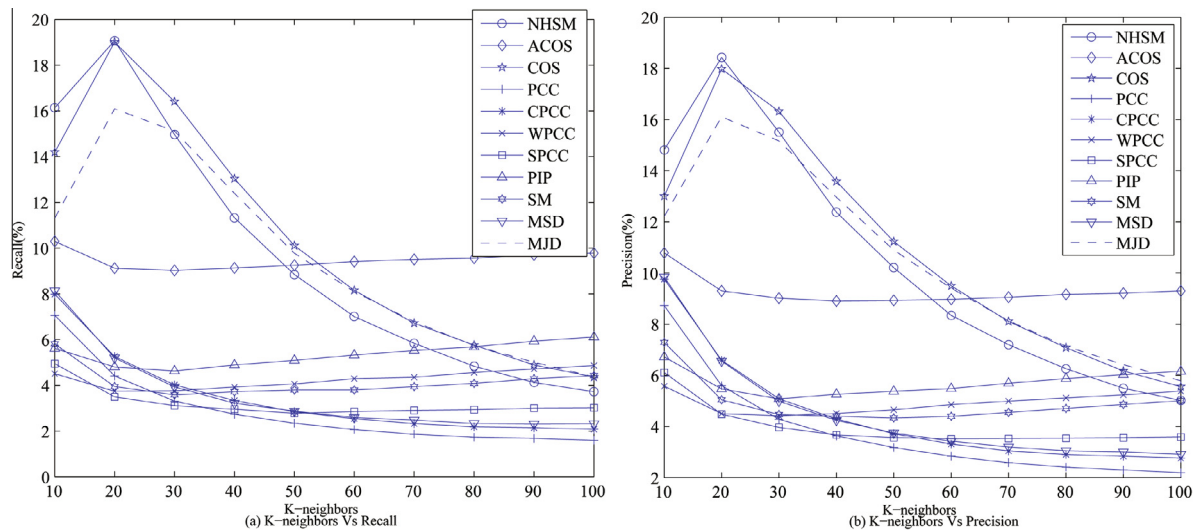


Fig. 5. The performances of different similarity measures with different K-neighbors on ML-1M data set.

becomes the best when the  $K$  is more than the 60. The most improvement of NHSM can reach about 40% compared with the MJD when the  $K$  is 10.

In Fig. 5(b), our improved similarity measure (NHSM) can obtain the best precision when the  $K$  is less than 20. Except the NHSM, the COS and MJD can get better performance than others. Compare with these two measures, the NHSM measure has 15.4% improvement when the  $K$  is 10. However, the PIP and other new similarity measure (SM) have a bad performance. However, the PIP is still better than PCC, WPCC, SPCC, CPCC and MSD when the  $K$  is more than 30.

Fig. 6 gives the comparison on Epinions data set. We can see that our proposed method is not the best in recall and precision when the  $K$  is small. However, the proposed method can obtain the best performance when the number of neighbor  $K$  is more than 80.

From Figs. 4–6, we can note that our improved similarity measure (NHSM) can obtain better performance with different numbers of neighbors than most other methods.

#### 4.4.2. The performance with different recommended lists

In top- $N$  recommendation, different numbers of recommendations will lead different performances. In this section, we use as the number of recommendations and analyze the impact of similarity measure with different  $TopN$ .

Fig. 7 shows the performances of all kind of similarity measures with different  $TopN$  on the ML-100K data set. From the figure we see that our improved similarity measure (NHSM) can obtain the best performance except cosine in the whole  $TopN$  range and the improvement is very remarkable compared with PIP.

In Fig. 7(a), we notice that the recalls of all the similarity measure will increase with the increasing of the number of recommendations. However, the NHSM and cosine increases faster than others. Except the NHSM and cosine, the MJD similarity is better than others. However, the recall of the NHSM can improve 20% compared with the MJD. The WPCC is the worst. We also note that other two new similarity measures, MSD and SM, which are better than the PCC, and PIP. However, they are worse than the COS and ACOS.



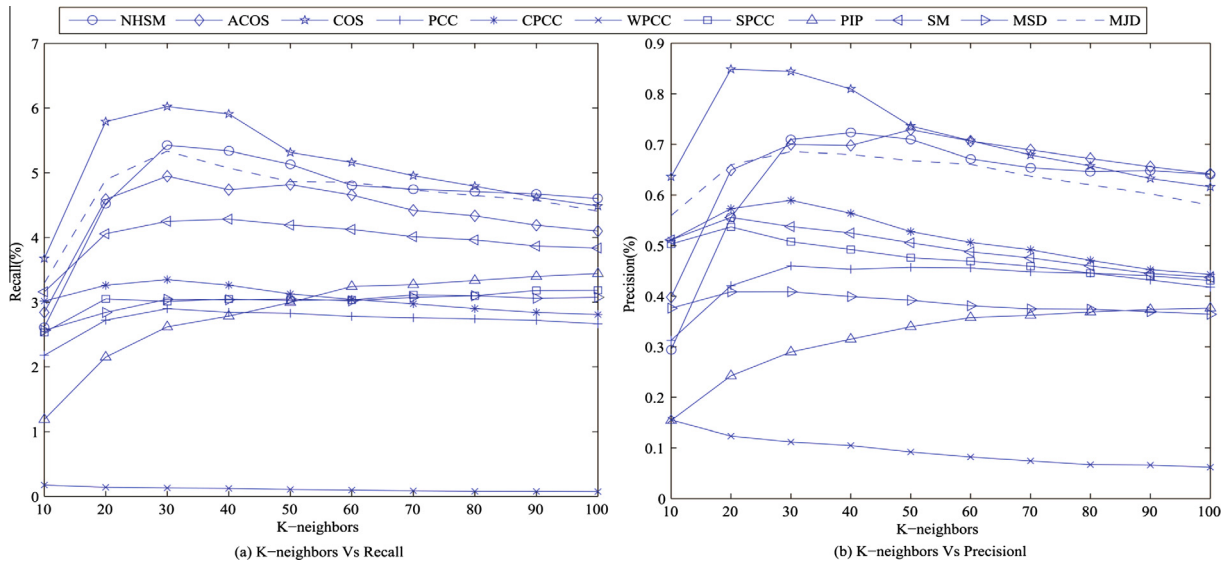


Fig. 6. The performances of different similarity measures with different K-neighbors on Epinions data set.

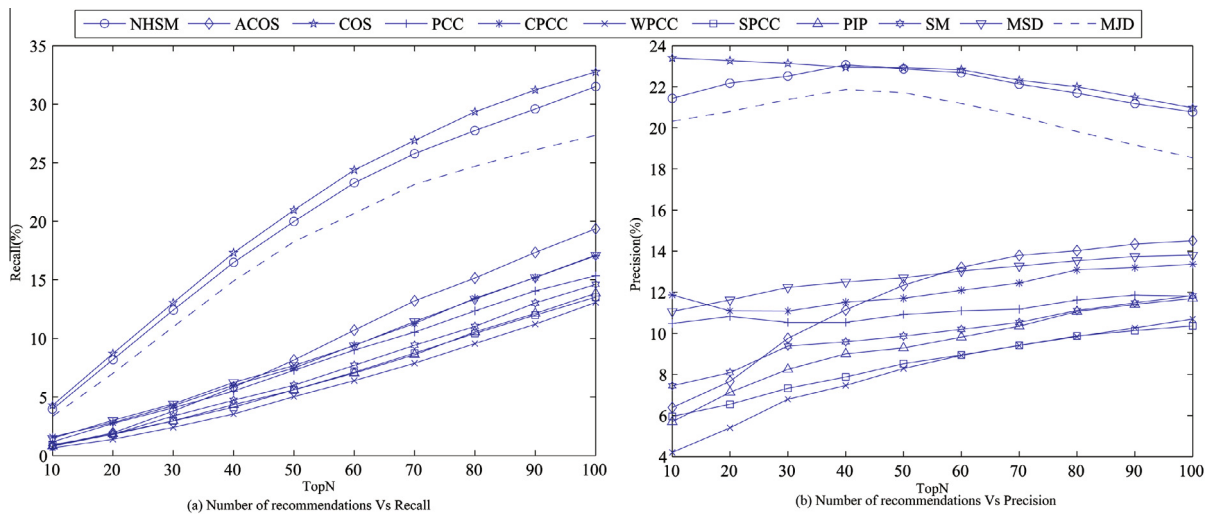


Fig. 7. The performances of different similarity measures with different TopN on ML-100K data set. We notice that our improved similarity measure (NHSM) has the best performance in the whole TopN range compared with these similarity measures.

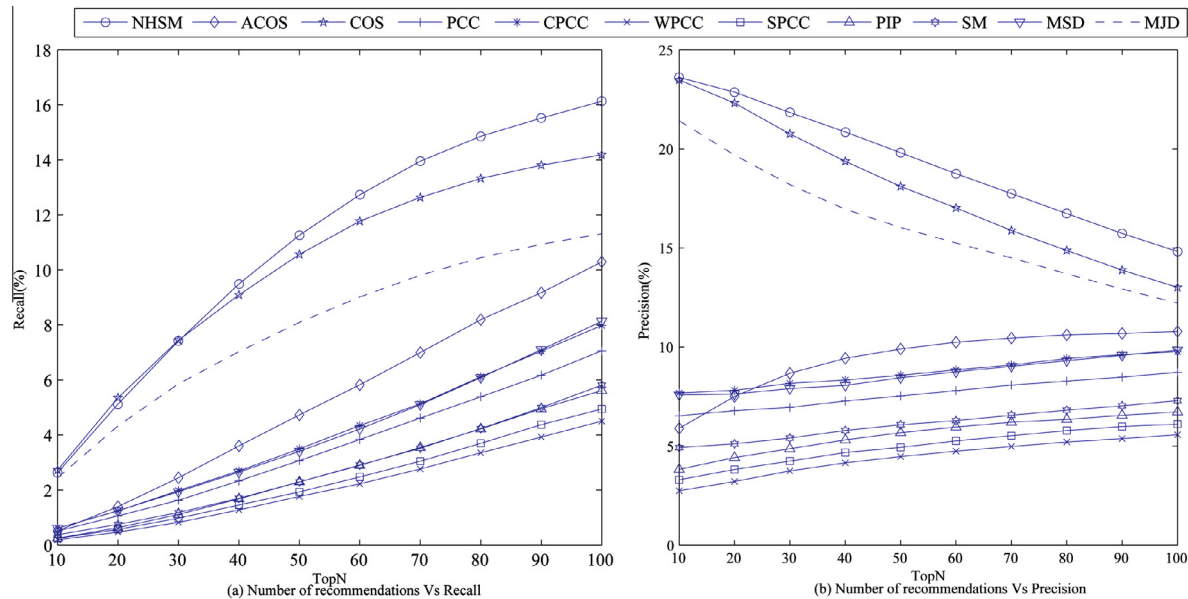
In Fig. 7(b), the precisions of all kind of similarity measures have no remarkable change in the whole TopN range. Except the NHSM, the COS is better than others when K is 40. We see that the PIP is better than WPCC and SPCC. Other two new similarity measures, SM and MJD, which can surpass the PCC, but they are worse than COS and NHSM.

Fig. 8 shows the performances of all similarity measures with different numbers of recommendations on the ML-1M data set. From the figure we see that the recall of all kind of similarity measures will increase with the increasing of the TopN. However, our improved similarity measure (NHSM) can also obtain the best performance in the whole range.

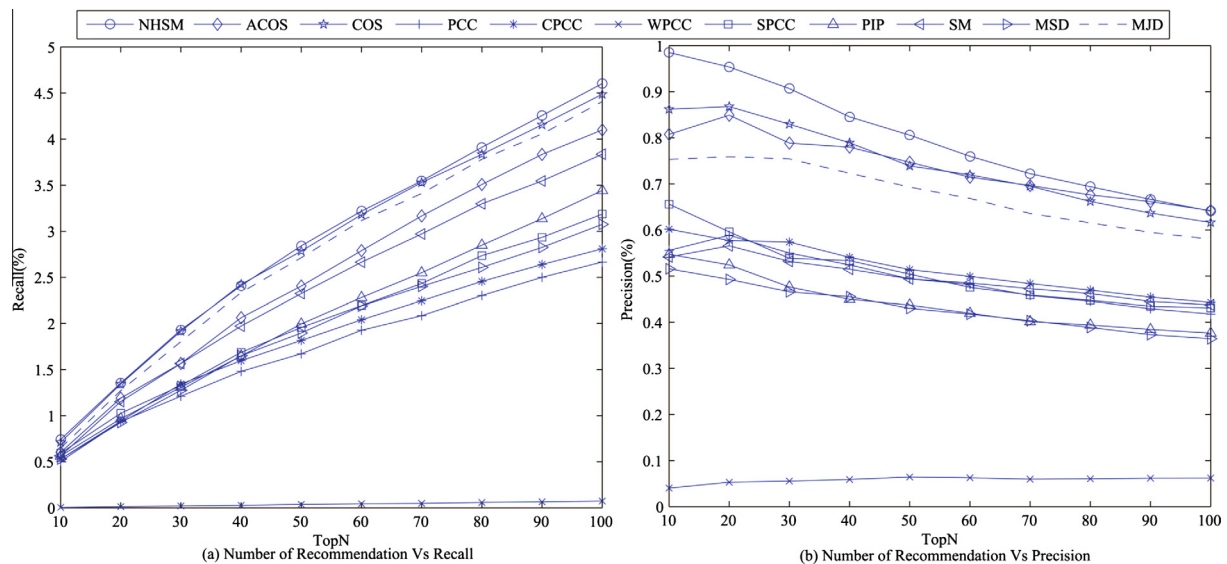
In Fig. 8(a), we notice that the improvement of the NHSM becomes larger with the larger TopN. Except the NHSM, we also see that the new similarity measure, MJD, which can better than others except COS. Compared with COS, the NHSM can improve 8.2% when the TopN is set 60. The PIP measure can surpass the SPCC and WPCC. However, compared with the PIP, the NHSM has a

remarkable improvement, it is about 300% when the TopN is set 60. The SM measure is similar with PIP and it is worse than the COS. The variant of cosine, ACOS, which is even worse than COS in the whole range of TopN. However, the variants of PCC, CPCC which have better results than PCC and SPCC and WPCC is worse than PCC.

In Fig. 8(b), the similarity of MJD and COS can surpass others in the whole TopN range except the NHSM measure. We notice that the improvement of NHSM has not very remarkable change compared with MJD and COS in the whole TopN. The PIP measure can surpass SPCC and WPCC in the whole range of TopN. However, the NHSM still has remarkable improvement compared with the PIP. This demonstrates that our improved similarity measure can have a better performance than PIP. Another new similarity measure, SM, which can have worse precision than PCC. Similar to the Fig. 8(a), the variant of cosine, ACOS, which has a worse precision than COS in the whole TopN range and the variants of PCC, CPCC which can surpass the PCC in the whole range of TopN.



**Fig. 8.** The performances of different similarity measures with different *TopN* on ML-1M data set. We notice that our improved similarity measure (NHSM) has the best performance in the whole *TopN* range compared with these similarity measures.



**Fig. 9.** The performances of different similarity measures with different *TopN* on Epinions data set. We notice that our improved similarity measure (NHSM) has the best performance in the whole *TopN* range compared with these similarity measures.

Fig. 9 shows the comparison on Epinions data set. Similarly, the proposed method can obtain the best performance. However, the superiority is not obvious compared with MovieLens. This is because the Epinions data set is too sparse. We notice that all methods are worse than MovieLens.

From Figs. 4–9, we can conclude that our improved similarity measure (NHSM) can obtain the better performance than most other methods. Second, the NHSM can surpass most of the state-of-the-art similarity measures in the whole number of recommendations range.

In one word, our improved similarity measure can obtain better performance than most other measures. These experiments demonstrate the effectiveness of our improved similarity measure model. Our measure can provide better performance that is because it captures the similarities between users and distinguishes them. Moreover, our model not only considers the local context

information of user ratings, but also takes the global information into account, such as the proportion of common rating, user rating preference.

## 5. Conclusions

The paper first analyzes the disadvantages of the existing similarity measures. In order to overcome these shortages, a novel similarity measure approach is proposed, which is based on the PIP measure. The initial PIP similarity is not normalized and the computing is complex. Hence, the paper proposes a new similarity model to overcome these shortages. Moreover, the improved similarity measure takes the proportion of the common rating between two users into account. Considering different users have different rating preferences, the paper uses the mean and variance

of the rating to describe the rating preference of user. For demonstrating the effectiveness of the novel similarity measure, several experiments are conducted on three popular used data sets. From the experimental results, we see that the novel similarity measure can obtain the better performance than most other methods. These results demonstrate the effectiveness of the novel similarity measure and it can overcome the drawbacks of the traditional similarity measures.

## Acknowledgments

This work is partially supported by the National 3rd Key Program project (No. 2011ZX03005-004-02), National Natural Science Foundation of China for “Social Search Research Oriented to Collaborative Generating Service” (No. 61302077), Key Project of the National Research Program of China (No. 2012BAH41F00, 2012BAH41F03), Funds for Creative Research Groups of China (61121001) and Program for Changjiang Scholars and Innovative Research Team in University (IRT1049).

## Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at <http://dx.doi.org/10.1016/j.knosys.2013.11.006>.

## References

- [1] N. Zheng, L. Qjudan, L. Shengcai, Z. Leiming, Which photo groups should I choose? A comparative study of recommendation algorithms in Flickr, *J. Inform. Sci.* 36 (6) (2010) 732–750.
- [2] E. Brynjolfsson, Y.J. Hu, M.D. Smith, Consumer surplus in the digital economy: estimating the value of increased product variety at online booksellers, *Manage. Sci.* 49 (11) (2003) 1580–1596.
- [3] B. Shummet, R. Seth, D. Sivakumar, Y. Jing, J. Yagnik, S. Kumar, D. Ravichandran, M. Aly, Video suggestion and discovery for YouTube: taking random walks through the view graph, in: *International Conference on World Wide Web*, 2008, pp. 895–904.
- [4] X. Zhang, Y. Li, Use of collaborative recommendations for web search: an exploratory user study, *J. Inform. Sci.* 34 (2) (2008) 145–161.
- [5] J.S. Breese, D. Heckerman, C. Kadie, Empirical analysis of predictive algorithms for collaborative filtering, in: *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, 1998, pp. 43–52.
- [6] F. Cacheda, V. Carneiro, D. Fernández, V. Formoso, Comparison of collaborative filtering algorithms: limitations of current techniques and proposals for scalable, high-performance recommender system, *ACM Trans. Web* 5 (1) (2011) 1–33.
- [7] M.J. Pazzani, D. Billsus, Content-based recommendation systems, *The Adap. Web* (2007) 325–341.
- [8] H. Junming, C. Xueqi, G. Jiafeng, S. Huawei, Y. Kun, Social recommendation with interpersonal influence, *ECAI* 10 (2010) 601–606.
- [9] L. Jie, S. Qusai, X. Yisi, L. Qing and Z. Guangquan, BizSeeker: a hybrid semantic recommendation system for personalized government-to-business e-services, *Internet Res.* 20(3) (2010) 342–365.
- [10] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, J. Riedl, GroupLens: an open architecture for collaborative filtering of netnews, in: *Proceeding of the ACM Conference on Computer Supported Cooperative Work*, 1994, pp. 175–186.
- [11] G. Adomavicius, A. Tuzhilin, 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) (2005) 734–749.
- [12] M. Ye, P. Yin, W.C. Lee, Location recommendation for location-based social networks, in: *Proceedings of the SIGSPATIAL International Conference on Advance in Geographic Information Systems*, 2010, pp. 458–461.
- [13] Q. Liu, T. Chen, J. Cai, D. Yu, Enlister: baidu's recommender systems for the biggest Chinese Q&A website, in: *Proceedings of the ACM Conference on Recommender Systems*, 2012, pp. 285–288.
- [14] H.J. Ahn, A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem, *Inform. Sci.* 178 (1) (2008) 37–51.
- [15] M. Jamali, M. Ester, TrustWalker: a random walk model for combining trust-based and item-based recommendation, in: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009, pp. 397–406.
- [16] J.L. Herlocker, J.A. Konstan, A. Borchers, J. Riedl, An algorithmic framework for performing collaborative filtering, in: *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1999, pp. 230–237.
- [17] G. Koutrica, B. Bercovitz, H. Garcia, FlexRecs: expressing and combining flexible recommendations, in: *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2009, pp. 745–758.
- [18] J. Bobadilla, F. Ortega, A. Hernando, J. Bernal, A collaborative filtering approach to mitigate the new user cold start problem, *Knowledge-Based Syst.* 26 (2011) 225–238.
- [19] J. Bobadilla, F. Ortega, A. Hernando, A collaborative filtering similarity measure based on singularities, *Inform. Process. Manage.* 48 (2012) 204–217.
- [20] J. Bobadilla, A. Hernando, F. Ortega, A. Gutierrez, Collaborative filtering based on significances, *Inform. Sci.* 185 (2012) 1–17.
- [21] D. Anand, K.K. Bharadwaj, Utilizing various sparsity measures for enhancing accuracy of collaborative recommender systems based on local and global similarities, *Expert Syst. Appl.* 38 (2011) 5101–5109.
- [22] M. Hao, I. King, R.I. Michael, Effective missing data prediction for collaborative filtering, in: *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2007, pp. 39–46.
- [23] A. Abdelwahab, H. Sekiya, I. Matsuba, Y. Horiuchi, S. Kuroiwa, Collaborative filtering based on an iterative prediction method to alleviate the sparsity problem, in: *Proceedings of the International Conference on Information Integration and Web-based Applications & Services*, 2009, pp. 375–379.
- [24] K. Goldberg, T. Roeder, G. Gupta, C. Perkins, Eigentaste: a constant time collaborative filtering algorithm, *Inform. Retrieval* 4 (2001) 133–151.
- [25] B.M. Sarwar, G. Karypis, J.A. Konstan, J. Riedl, Application of dimensionality reduction in recommender system – a case study, in: *ACM WebKDD Workshop*, 2000.
- [26] G. Song Jie, Y. Hong Wu, D. YaE, Combining singular value decomposition and item-based recommender in collaborative filtering, in: *International Workshop on Knowledge Discovery and Data Mining*, 2009, pp. 769–772.
- [27] A. Szwabe, M. Ciesielczyk, T. Janasiewicz, Semantically enhanced collaborative filtering based on RSVD, in: *Proceedings of the International Conference on Computational Collective Intelligence*, 2011, pp. 10–19.
- [28] M. Hao, Y. Haixuan, M.R. Lyu, K. Irwin, Sorec: social recommendation using probabilistic matrix factorization, in: *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, 2008, pp. 931–940.
- [29] X. Gui Rong, L. Chenxi, Y. Qiang, X. Wensi, Z. Hua Jun, Y. Yong, C. Zheng, Scalable collaborative filtering using cluster-based smoothing, in: *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2005, pp. 114–121.
- [30] M. Grcar, D. Mladenic, B. Fortuna, M. Grobelnik, Data sparsity issues in the collaborative filtering framework, *Advan. Web Min. Web Usage Anal.* (2006) 58–76.
- [31] C. DanEr, The collaborative filtering recommendation algorithm based on BP neural networks, in: *Proceedings of the International Symposium on Intelligent Ubiquitous Computing and Education*, 2009, pp. 234–236.
- [32] L. Nan, L. Chunping, Zero-sum reward and punishment collaborative filtering recommendation algorithm, in: *Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence Agent Technology*, 2009, pp. 548–551.
- [33] U. Shardanand, P. Maes, Social information filtering: algorithms for automating word of mouth, in: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1994, pp. 210–217.
- [34] P. Cremonesi, Y. Koren, R. Turrin, Performance of recommender algorithms on top-n recommendation tasks, in: *Proceedings of the ACM Conference on Recommender Systems*, 2010, pp. 39–46.
- [35] F. Francois, P. Alain, R. Jean-Michel, S. Marco, Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation, *IEEE Trans. Knowl. Data Eng.* 19 (2007) 355–369.
- [36] Y. Seok Jong, The dynamic competitive recommendation algorithm in social network services, *Inform. Sci.* 187 (2012) 1–14.