RESEARCH ARTICLE

An improved collaborative filtering method based on similarity

Junmei Feng[1]*, **Xiaoyi Fengs**[1], **Ning Zhang**[1], **Jinye Peng**[2]

1 School of Electronics and Information, Northwestern Polytechnical University, Xi'an, Shaanxi, China,
2 School of Information Science and Technology, Northwest University, Xi'an, Shaanxi, China

* 18710993442@163.com

Citation: Feng J, Fengs X, Zhang N, Peng J (2018) An improved collaborative filtering method based on similarity. PLoS ONE 13(9): e0204003. https://doi.org/10.1371/journal.pone.0204003

Editor: Hua Wang, Victoria University, AUSTRALIA

Received: November 13, 2017

Accepted: August 31, 2018

Published: September 24, 2018

Data Availability Statement: Movielens 100K is publicly available at http://grouplens.org/datasets/movielens. FilmTrust and Ciao are publicly available at https://www.librec.net/datasets.html. Epinions publicly available at http://www.trustlet.org/datasets/.

Funding: This work is supported by the Changjiang Scholars and Innovative Research Team in University Ministry of Education of China (number IRT_17R87, http://www.nsfc.gov.cn/, decision to publish and preparation of the manuscript).

Competing interests: The authors have declared that no competing interests exist.

Abstract

The recommender system is widely used in the field of e-commerce and plays an important role in guiding customers to make smart decisions. Although many algorithms are available in the recommender system, collaborative filtering is still one of the most used and successful recommendation technologies. In collaborative filtering, similarity calculation is the main issue. In order to improve the accuracy and quality of recommendations, we proposed an improved similarity model, which takes three impact factors of similarity into account to minimize the deviation of similarity calculation. Compared with the traditional similarity measure, the advantages of our proposed model are that it makes full use of rating data and solves the problem of co-rated items. To validate the efficiency of the proposed algorithm, experiments were performed on four datasets. Results show that the proposed method can effectively improve the preferences of the recommender system and it is suitable for the sparsity data.

1 Introduction

With the rapid development of internet technology, the amount of information on the internet grows exponentially. To overcome the information overload problem [1], the recommender system (RS) [2, 3] has been widely used in our daily life to recommend information of the customer's interest or provide personalized services based on the customer's behavior data, which help the customer quickly obtain the required information from the mass data. Recently, RS has been successfully applied to a variety of fields, such as music [4], TV show [5], E-commerce [6], news [7], taxi [8], tourism [9], social media [10].

In RS, recommendations are made through different types of algorithms. As the core and key part of the RS, the recommendation algorithm [11, 12] determines the type and performance of the RS. In general, the recommendation algorithm is classified into four major categories:

- Demographic filtering approach [13], which is based on the assumption that customers with similar personal attributes (age, sex, etc.) may have similar preferences. The calculation is simple, and it is easy to achieve real-time response. Since the preferences of different customers with common personal attributes may be different, one of main problems of this method is the low reliability.

- Content-based filtering approach [14, 15], which utilizes user's choices made in the past. Therefore, items that are similar to what the user has previously purchased or liked will be recommended.
- Collaborative filtering (CF) approach [16], where recommendations are made based on the user's ratings of the items. Users with similar ratings are called nearest neighbors, if the nearest neighbors are found, the unrated items of the user are predicted through the neighbors, then, the RS recommends the items with high predicted ratings to the user.
- Hybrid filtering approach [17] which combines previous approaches using different knowledge sources to solve the problems existing in each one of these algorithms.

Among these recommendation approaches, CF is generally considered as one of the most used and most successful recommendation technologies in the RS, especially e-commerce websites such as Amazon.com, Netflix and Google News [18].

Collaborative filtering recommendation algorithms are usually classified into two classes: memory-based algorithms [19, 20] and model-based algorithms [21, 22]. The main difference is the processing of ratings. Memory-based algorithms include user-based collaborative filtering (UBCF) algorithms [20] and item-based collaborative filtering (IBCF) algorithms [23]. The UBCF algorithm focuses on obtaining the target user's nearest neighbors and predicting his/her unrated items, conversely, the goal of the IBCF algorithm is items. In this paper, UBCF is taken to illustrate the improved method of similarity. The model-based algorithm [24] needs to build a model that represent users' behavior according to the collected ratings, then, the unrated items can be predicted.

The value of similarity in CF mentioned above needs a similarity function to measure, some commonly used similarity metrics including cosine (COS), Pearson Correlation Coefficient (PCC), weighted Pearson Correlation Coefficient (WPCC) and Jaccard. Once the similarity method is chosen, recommendations can be made to users. Generally speaking, the above measures can well reflect the degree of similarity between two users or items. However, when the dataset is sparse, the accuracy of recommendation is very low. To solve this problem, plenty of similarity metrics have been proposed in recent years, nevertheless, the improvement is not obvious.

In this paper, our goal is to devise a similarity method that works for most recommender systems, regardless of the sparsity of the datasets. Based on the above considerations, this paper proposes three similarity impact factors to improve the accuracy and quality of the recommendations. Furthermore, the proposed similarity algorithm is normalized.

The structure of the paper is as follows. Section 2 reports some similarity calculation methods in the field of collaborative filtering recommendation. Section 3 describes the proposed similarity model in detail. Section 4 presents the experimental results in different datasets. Section 5 discusses the results and advantages of our proposed model. Finally, Section 6 gives a high level description of the conclusions and future work.

2 Related work

In this section, we briefly summarize the related work of similarity metrics. The larger the value, the higher the correlation. In the following formulas, we assume that the set of users and items are $U = \{u_1, u_2, \cdots, u_m\}$ and $I = \{i_1, i_2, \cdots, i_n\}$ respectively. $R = [r_{ui}]^{m\times n}$ is used to represent the user-item rating matrix. Here, the number of users and items are m and n respectively, r_{ui} represents the rating made by user u on item i.

COS [25] similarity measures the angle between two rating vectors (users or items). Its similarity is frequently used in CF recommender system. The formula of COS similarity between user u and v is defined in Eq (1):

$$\mathrm{Sim}(u,v)^{\mathrm{COS}} = \frac{\vec{r}_u \cdot \vec{r}_v}{\|\vec{r}_u\| \cdot \|\vec{r}_v\|} = \frac{\sum_{i\epsilon I_u \cap I_v} r_{ui} \cdot r_{vi}}{\sqrt{\sum_{i\epsilon I_u \cap I_v} r_{ui}^2} \cdot \sqrt{\sum_{i\epsilon I_u \cap I_v} r_{vi}^2}} \tag{1}$$

However, COS does not consider the user's rating preference. In other words, some users tend to score high in general, while others prefer to give low, even if they like the items very much. Adjust cosine (ACOS) [26] similarity measure solves this problem by subtracting the average rating.

PCC [27] is defined on the set of co-rated items or users. WPCC [28] is based on PCC. The formulas of PCC and WPCC are described in Eqs (2) and (3) respectively. Intuitively, when the number of the co-rated items is less than the threshold in Eq (3), the similarity value is smaller than the result of PCC. On the contrary, if the number is the threshold or more, the similarity metric is still PCC. In experiments, the threshold is usually set to 50. Constrained Pearson Correlation (CPCC) [29] is a modified form of PCC, in which an absolute reference is used instead of the average rating. When the co-rated values are on the same side, the correlation can be increased. Another modified form of PCC is sigmoid function based on Pearson Correlation Coefficient (SPCC) [30] that weakens the similarity compared with PCC.

$$\mathrm{Sim}(u,v)^{\mathrm{PCC}} = \frac{\sum_{i\epsilon I_u \cap I_v} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i\epsilon I_u \cap I_v} (r_{ui} - \bar{r}_u)^2} \cdot \sqrt{\sum_{i\epsilon I_u \cap I_v} (r_{vi} - \bar{r}_v)^2}} \tag{2}$$

$$\mathrm{Sim}(u,v)^{\mathrm{WPCC}} = \begin{cases} \dfrac{|I_u \cap I_v|}{T} \cdot sim(u,v)^{PCC}, & if\ \ |I_u \cap I_v| < T \\ sim(u,v)^{PCC}, & otherwise \end{cases} \tag{3}$$

In addition, Jaccard [31] is another popular measure used in CF. This measure only considers the number of items rated by two users instead of the ratings, which indicates the more co-rated items, the more similar. Therefore, the similarity metric is inaccurate in some cases. Different from Jaccard, Mean Squared Difference (MSD) [20] considers more about the absolute ratings. However, the application of this measure is not very wide. The formulas of Jaccard and MSD is shown in Eqs (4) and (5) respectively.

$$\mathrm{Sim}(u,v)^{\mathrm{Jaccard}} = \frac{|I_u \cap I_v|}{|I_u \cup I_v|} \tag{4}$$

$$\mathrm{Sim}(u,v)^{\mathrm{MSD}} = 1 - \frac{\sum_{i\epsilon |I_u \cap I_v|} (r_{ui} - r_{vi})^2}{|I_u \cap I_v|} \tag{5}$$

To avoid the drawbacks of traditional measures, in [32], Bobadilla et al. came up with a method that combined Jaccard and Mean Squared Difference (JMSD), in which Jaccard is used to capture the proportion of the co-rated items and MSD is used to obtain the information of ratings. The formula of JMSD is expressed in Eq (6).

$$\mathrm{Sim}(u,v)^{\mathrm{JMSD}} = Sim(u,v)^{Jaccard} \cdot Sim(u,v)^{MSD} \tag{6}$$

In [33], Bobadilla et al. proposed another similarity method named MJD (Mean-Jaccard-Differences), which combined six similarity measures to obtain a global similarity. The weight of each measure was obtained through neural network learning. However, these two measures do not work in the case of sparse data.

Another classical method proposed by Ahn used three factors of similarity, namely Proximity, Impact and Popularity called PIP (Proximity-Impact-Popularity) [26]. Although PIP can alleviate the cold start problem, the disadvantages are still obvious. First, the similarity metric does not consider the absolute ratings, and it also ignores the proportion of the co-rated items. Second, the method does not consider each user's global rating preference. Finally, the formula is not normalized and it is not convenient to combine with other methods. Based on the above considerations, Liu et al. proposed a new heuristic similarity model (NHSM) in [20]. This method is based on PIP and successfully overcomes the inadequacies of the PIP approach. The formula of NHSM is expressed in Eq (7).

$$\mathrm{Sim}(u,v)^{\mathrm{NHSM}} = Sim(u,v)^{JPSS} \cdot Sim(u,v)^{URP} \tag{7}$$

In [34], Polatidis et al. proposed a multi-level recommendation method to improve the quality of RS. This measure divides similarity into different levels and adds constrains to each level, the final similarity value depends on PCC and the number of co-rated items. The similarity metric adds a different constant to different level. The more co-rated items, the greater the constant.

Patra et al. proposed a new similarity measure using Bhattacharyya coefficient memory-based CF in sparse data, which used all ratings made by a pair of users in [35]. Beyond that, Zhang et al. proposed a novel data structure and designed linear algorithms to compute the similarities in [36], the final goal is to short the evaluation time and improve the efficiency of the development of RS. Moreover, Lee et al. introduced a preference model in [37], which is used to improve the accuracy of all existing CF algorithms. The preference model is obtained by maximum likehood estimation. On a recent work, Sun et al. proposed a new similarity measure of Triangle Multiplying Jaccard (TMJ) in [38], which combines triangle similarity and Jaccard similarly to improve recommendation accuracy. The TMJ is defined in Eq (8).

$$\mathrm{Sim}(u,v)^{\mathrm{TMJ}} = Sim(u,v)^{Triangle} \cdot Sim(u,v)^{Jaccard} \tag{8}$$

In summary, literature offers rich evidence on the successful performance of CF measures. However, the existing similarity method still has some limitations. First, CF measures suffer from serious data sparsity [39] and cold start [40] problems. In practice, the user-item rating matrix used for CF is extremely sparse and does not have enough ratings, so the performance of the CF recommender system is challenged by data sparsity. Cold start problem is an extreme situation that occurs when a new user or item just enters the system, and it is difficult to recommend for the lack of information. In the aforementioned issues, This paper focuses on the data sparsity problem. The main contribution of our work is that we propose a novel similarity model to minimize the deviation of similarity calculation and improve the accuracy of the recommendations, and our model can still maintain high recommendation accuracy in the case of data sparsity.

3 The proposed similarity model

This section first introduce the motivations of our proposed similarity model. Then, we give a description of the proposed similarity model in detail and analyze its time complexity. Finally, we present the prediction measure adapted in our work.

3.1 The motivations of the proposed similarity model

From the description of the previous section, we notice that the traditional CF methods heavily rely on the co-rated items. However, the similarity computation cannot be performed when there are no co-rated items, which is called co-rated items problem. Therefore, our novel model is proposed to solve this situation.

The motivations for our proposed similarity model lies in three aspects: First, our model takes all rated items into consideration, while, the traditional CF approaches only considers the co-rated items, which accounts for a small fraction of the rated items. Second, the proposed model can solve the co-rated items problem in datasets, even for the extremely sparse datasets. Third, the similarity model is not only decided by all the rated items, but also the user's global preference.

3.2 Proposed algorithm

The recommendation algorithm in this paper provide users with recommendations through three steps: Initially, the ratings generated by the behavior of a user's interactions are extracted and stored to the database. Then, the approach of k-nearest neighbors (KNN) [41] is applied to predict the ratings of the target user's unrated items. The difficulty of KNN is how to calculate the similarities between the target user and his/her neighbors. An improved similarity model is proposed to minimize the deviation of similarity calculation and improve the accuracy of recommendation, which will be introduced below. Finally, the first N items with the top predicted ratings will be recommended to the target user.

The main part of memory-based CF method is similarity calculation, which can be calculated either on pair of users or items. To evaluate the proposed similarity algorithm, UBCF is adapted in this paper. In order to improve the adaptability of the similarity metric in the case of the sparse rating data, the proposed similarity model is composed of three impact factors including S_1, S_2 and S_3. Additionally, S_1 is used to define the similarity between users. S_2 is introduced to punish the user pairs with small proportion of the number of co-rated items. S_3 is adopt to weight each user's rating preference. The framework is defined in Eq (9).

$$\mathrm{Sim}(u,v)^{\mathrm{Proposed}} = S_1(u,v)\cdot S_2(u,v)\cdot S_3(u,v) \tag{9}$$

The similarity S_1 is defined to measure the angle between the rating vectors of two users. The smaller the angle, the higher the degree of the similarity between the two users. Different from the traditional COS similarity method, S_1 converts the angle calculation problem from the original $|I_u \cap I_v|$ dimension space to $|I_u \cup I_v|$ dimension space. That is, the calculation converts from the set of two users' co-rated items to the union of two users' rated items, which makes the rating data fully utilized. However, the sparsity of the dataset in the RS directly determines the accuracy of the angle calculation. If the sparsity of the dataset is low, the similarity between two users is calculated based on all existing rating data. In contrast, if the sparsity is high, there is almost no co-rated items between any two users, and the traditional similarity method does not work. In this case, we construct co-rated items in the new rating space, and replace the ratings of the unrated items with the average ratings to improve the accuracy of the algorithm. Based on the sparsity of dataset, S_1 is divided into to two levels. The formula of S_1 is defined in Eq (10). From the formula, it can be seen that the denominator has become larger compared with the traditional COS measure.

$$\begin{aligned} S_1(u,v) &= \begin{cases} \dfrac{\sum_{i\epsilon I_u\cap I_v} r_{ui}\cdot r_{vi}}{\sqrt{\sum_{i\epsilon I_u} r_{ui}^2}\cdot\sqrt{\sum_{i\epsilon I_v} r_{vi}^2}} & \mathit{if}\ \ \mathit{sparsity} < \rho \\ \dfrac{\sum_{i\epsilon I_u\cup I_v} r_{ui}\cdot r_{vi}}{\sqrt{\sum_{i\epsilon I_u\cup I_v} r_{ui}^2}\cdot\sqrt{\sum_{i\epsilon I_u\cup I_v} r_{vi}^2}} & \mathit{otherwise} \end{cases} \\ &= \begin{cases} \dfrac{\sum_{i\epsilon I_u\cap I_v} r_{ui}\cdot r_{vi}}{\sqrt{\sum_{i\epsilon I_u} r_{ui}^2}\cdot\sqrt{\sum_{i\epsilon I_v} r_{vi}^2}} & \mathit{if}\ \ \mathit{sparsity} < \rho \\ \dfrac{\sum_{i\epsilon I_u\cap I_v} r_{ui}\cdot r_{vi} + \sum_{i\epsilon I_u - I_u\cap I_v} r_{ui}\cdot\mu_v + \sum_{i\epsilon I_v - I_u\cap I_v}\mu_u\cdot r_{vi}}{\sqrt{\sum_{i\epsilon I_u} r_{ui}^2 + \sum_{i\epsilon I_u\cup I_v - I_u}\mu_u^2}\cdot\sqrt{\sum_{i\epsilon I_v} r_{vi}^2 + \sum_{i\epsilon I_u\cup I_v - I_v}\mu_v^2}} & \mathit{otherwise} \end{cases} \end{aligned} \tag{10}$$

Where μ_u and μ_v are the average ratings of user u and user v respectively. In the case of sparsity less than the threshold ρ, the similarity is calculated in $|I_u \cup I_v|$ dimension space. In the new rating space, zero is used to indicate the user's rating of the unrated items. Unlike the traditional COS similarity method, which only uses the rating data on the co-rated items, our proposed similarity S_1 uses the two users' entire rating data. In other cases, the ratings of the unrated items in this space are replaced with the users' average ratings, and then calculate the similarity.

In the recommender system, the number of co-rated items between different user pairs varies greatly. The more co-rated items, the more valuable information is extracted from the rating data, and the similarity calculation result will be more accurate. Consequently, the proportion of the number of co-rated items is a very important impact factor, and its definition is in Eq (11). If the proportion of the co-rated items is small, the value of S_2 will be low.

$$S_2(u,v) = \frac{1}{1 + \exp\left(-\frac{(I_u\cap I_v)^2}{|I_u|\cdot|I_v|}\right)} \tag{11}$$

In our model, S_3 is used to indicate the rating preference of each user. Due to different users have different rating habits, some users like to give high ratings, while others may prefer rating low. Therefore, the rating preference should be considered. S_3 is adopted to revise our proposed model. The formula of S_3 is defined in Eq (12) [19], which is determined by average rating and standard variance.

$$S_3(u,v) = 1 - \frac{1}{1 + \exp\left(-|\mu_u - \mu_v|\cdot|\delta_u - \delta_v|\right)} \tag{12}$$

Where $|\cdot|$ returns the absolute value of the function. δ_u and δ_v represent the standard variance of user u and user v.

Thus, our proposed similarity model is divided into two levels based on the data sparsity of the RS, the similarity between user u and v is the product of S_1, S_2 and S_3. Compared with the traditional CF methods, the advantages of our proposed model are: First, more data not just the co-rated items is used in the similarity factor of S_1, to extract more useful information. Second, from the defined formulation of S_1, we notice that the proposed model completely solves the problem of co-rated items. Consequently, it broadens the scope of application of the traditional memory-based CF approaches, even for the sparse data. Third, we not only consider the influence the co-rated items (factor S_2), but also take into account the impact of the global preference of user's behavior (factor S_3).

3.3 Time complexity

According to the assumptions in section 2, the number of users and items are m and n respectively. From the definition of our proposed similarity model we can see that its time complexity of user similarity computation is $O(n)$. In this paper, KNN is adopted to find each user's nearest neighbors. Hence, the time complexity of finding all neighbors is $O(mn)$.

Since the maximum number of ratings in the dataset is mn, all unrated items should be predicted by the proposed model, therefore, the overall time complexity for the whole dataset is $O(m^2n^2)$.

3.4 Ratings prediction

If the similarities are prepared, the ratings of unrated items can be predicted. In this paper, the prediction formula [26] of a rating of user u on item i is expressed in Eq (13).

$$\mathrm{p}_{\mathrm{ui}} = \bar{r}_u + \frac{\sum_{v\epsilon NN_u} Sim(u,v)\cdot(r_{vi}-\bar{r}_v)}{\sum_{v\epsilon NN_u} |Sim(u,v)|} \tag{13}$$

Where NN_u indicates the set of nearest neighbors of user u. Based on the above method, all unrated items of the target user can be calculated. Then, the RS recommends the top N items to the target user as the recommendation results.

4 Experiments

In this section, we first carry out a set of sparsity experiments to determine the optimal threshold ρ in our proposed model. Then, to verify the superiority of the proposed similarity model based on CF, we conduct experimental evaluation on four real datasets and compare our model with other methods of CF including COS, PCC, WPCC, Jaccard, MSD, JMSD, NHSM and TMJ, which are described in the related work section.

4.1 Datasets

Four datasets of Movielens 100K, FilmTrust, Ciao and Epinions are employed to evaluate the effectiveness of our algorithm. Because these four datasets are often used by researchers to verify the performance of CF recommendation algorithms, and their sparsity is quite different, so we choose the above four datasets. In this paper, we assume that sparsity is the ratio of the number of unrated user-item pairs to the total number of user-item pairs in the user-item rating matrix. Moreover, the sparsity in Eq (10) refers to the sparsity of the training datasets.

- Movielens 100K dataset (http://grouplens.org/datasets/movielens) contains 100,000 ratings of 1,682 movies made by 943 users. In this dataset, each user has rated at least 20 movies. All the rating values are integer in the scale 1 to 5, where rate 1 shows that the user is not interested in the movie, and rate 5 means that the user favors the movie very much. The sparsity of Movielens 100K is 93.7%.
- FilmTrust dataset is a small and publicly available dataset extracted from the entire FilmTrust website. The dataset has 35,497 ratings from 1,508 users on 2,071 movies, and the rating value is a multiple of 0.5, ranging from 0.5 to 4. The sparsity of FilmTrust is 98.86%.

- Ciao dataset is a publicly available dataset retrieved from the entire DVD category on the dvd.co.uk website. It contains 72,665 ratings from 17,615 users on 16,121 movies and all the rating values are integer in the scale 1 to 5. The sparsity of Ciao is 99.9744%.
- Epinions dataset (http://www.trustlet.org/epinions.html) is a publicly available and general product recommendation dataset. The dataset contains 664,823 ratings of 139,738 items rated by 40,163 users. The rating range in the dataset is an integer value from 1 to 5. Moreover, the sparsity of Epinions is 99.9882%.

To demonstrate the performance of the proposed similarity measure, each dataset is divided into two parts, 80% randomly selected of part is used for training, and the remaining 20% is used for testing. Hence, the sparsity of the above four training datasets are 94.956%, 99.09%, 99.98%, and 99.99% respectively.

4.2 Evaluation metrics

To estimate the performance of RSs, the mean absolute error (MAE), rooted mean squared error (RMSE) [42], precision and recall are among the most popular ones. According to [43], the metrics evaluating recommendation systems can be roughly classified into two categories: prediction accuracy and classification accuracy. MAE and RMSE are mainly used to evaluate the prediction accuracy [36], while precision and recall are used to evaluate the quality of top-N recommendation [20]. In this paper, we adopt the metrics of MAE and RMSE to represent the accuracy of the proposed algorithm.

MAE is the most used metric in collaborative filtering RS, which is used to estimate the average absolute deviation between the actual ratings and the prediction ratings, MAE is defined in Eq (14).

$$\mathrm{MAE} = \frac{1}{TN} \sum_{u\epsilon U, i\epsilon I} |p_{ui} - r_{ui}| \tag{14}$$

Where TN is the total number of items in the set. p_{ui} and r_{ui} represent the predicted rating and actual rating of user u on item i respectively. The smaller the metric, the higher the prediction accuracy. However, MAE is not normalized.

RMSE reflects the degree of deviation between the predicted ratings and the actual ones, which penalizes large deviation more heavily for squaring the errors before summing them. Lower RSME corresponds to higher prediction accuracy. RMSE is evaluated in Eq (15).

$$\mathrm{RMSE} = \sqrt{\frac{1}{TN} \sum_{u\epsilon U, i\epsilon I} (p_{ui} - r_{ui})^2} \tag{15}$$

4.3 The threshold ρ

We remind that the threshold ρ is a policy factor affecting the results of the proposed model. Before the experimental evaluation process, we set the threshold in the proposed model to the extremes of $\rho = 1$ and $\rho = 0$ respectively, in which case the model is converted from two levels (level 1 and level 2) to one level. When $\rho = 1$, the remaining level is called level 1. Similarly, $\rho = 0$ corresponds to level 2. In this section, our research focuses on the effects of sparsity on the two different levels to further determine the optimal threshold. Moreover, the performance of the recommendation is estimated by the metrics of MAE and RMSE. In CF algorithms, since the number of nearest neighbors also affects the accuracy and quality of recommendation, it is set to a fixed value of 30 in this set of experiments on sparsity.

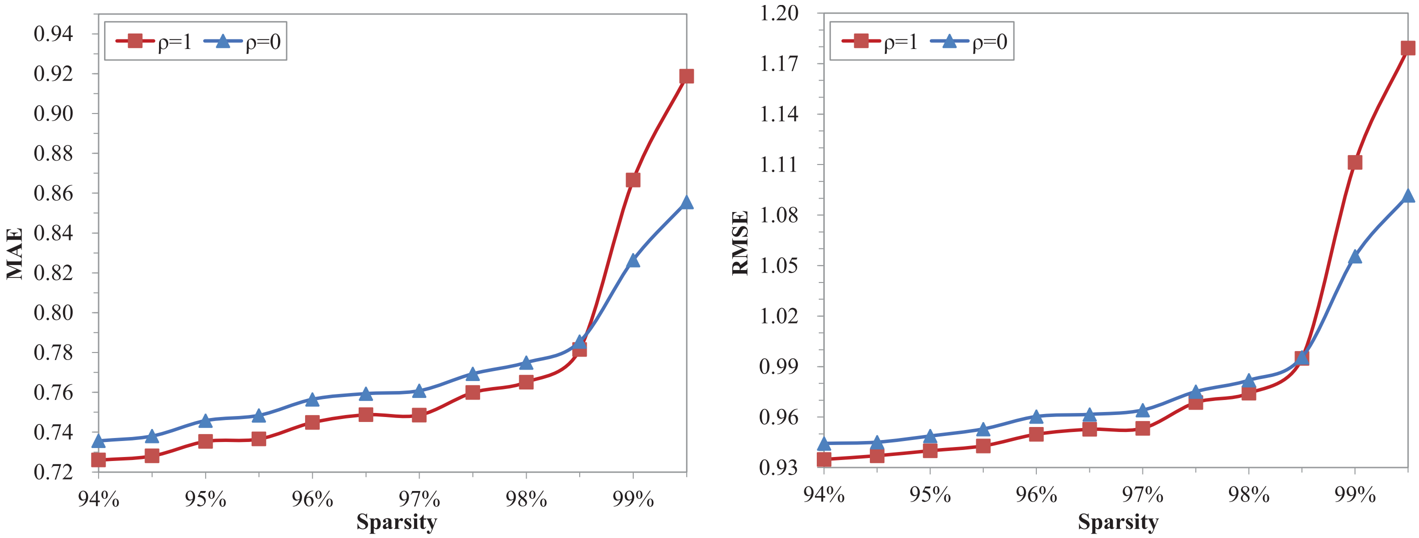

Fig 1. The performance of the proposed model at two extreme thresholds with different sparsity.

https://doi.org/10.1371/journal.pone.0204003.g001

In Fig 1, we illustrate our results of the proposed model on the conditions of $\rho = 1$ and $\rho = 0$ with the values of MAE and RMSE changed by sparsity. Datasets with different sparsity levels are constructed by changing the proportion of the training set in the Movielens 100K. Sparsity varies from 94% to 99.5%, and the step size is 0.5%. From the figure we see that the performance of our model continually deteriorates under two threshold conditions with the increasing of sparsity, especially when the sparsity is greater than 98.5%. It demonstrates that sparsity is an important factor affecting the accuracy of recommendation, and the higher the sparsity, the greater the impact. When the sparsity is less than 98.5%, our proposed model on the condition of $\rho = 1$ has lower MAE and RMSE, contrarily, the model on the condition of $\rho = 0$ has better metrics when the sparsity is greater than 99%. As $\rho = 1$ corresponds to the level 1, and $\rho = 0$ corresponds to the level 2, the figure clearly show that the level 1 surpasses the level 2 when the sparsity is less than 98.5%, however, the level 2 has better performance when the sparsity is greater than 99%. Consequently, we set the threshold $\rho = 0.985$ in the following experiments to ensure that the proposed model performs better in the whole sparsity range.

4.4 Settings

In order to evaluate the effectiveness of the proposed similarity model, we compare our similarity algorithm with some other measures by using the metrics of MAE and RMSE in four datasets. As is known, the performance of the recommendation algorithms is affected by the number of nearest neighbors, which is denoted by K in this paper. Considering the recommendation efficiency and accuracy of the recommendation algorithm, K varies from 5 to 100, and the step size is 5. Besides, the threshold ρ in Eq (10) is set to 0.985. For the below experiment evaluation, the experiment results of our proposed method are compared with the other eight CF recommendation algorithms under the conditions of different K.

4.5 Experimental results and analysis

Figs 2 and 3 show the MAE and RMSE of different similarity measures with different K on the dataset of Movielens 100K respectively. Both MAE and RMSE firstly decrease with the increasing of nearest neighbors and then increase slightly for different similarity measures except

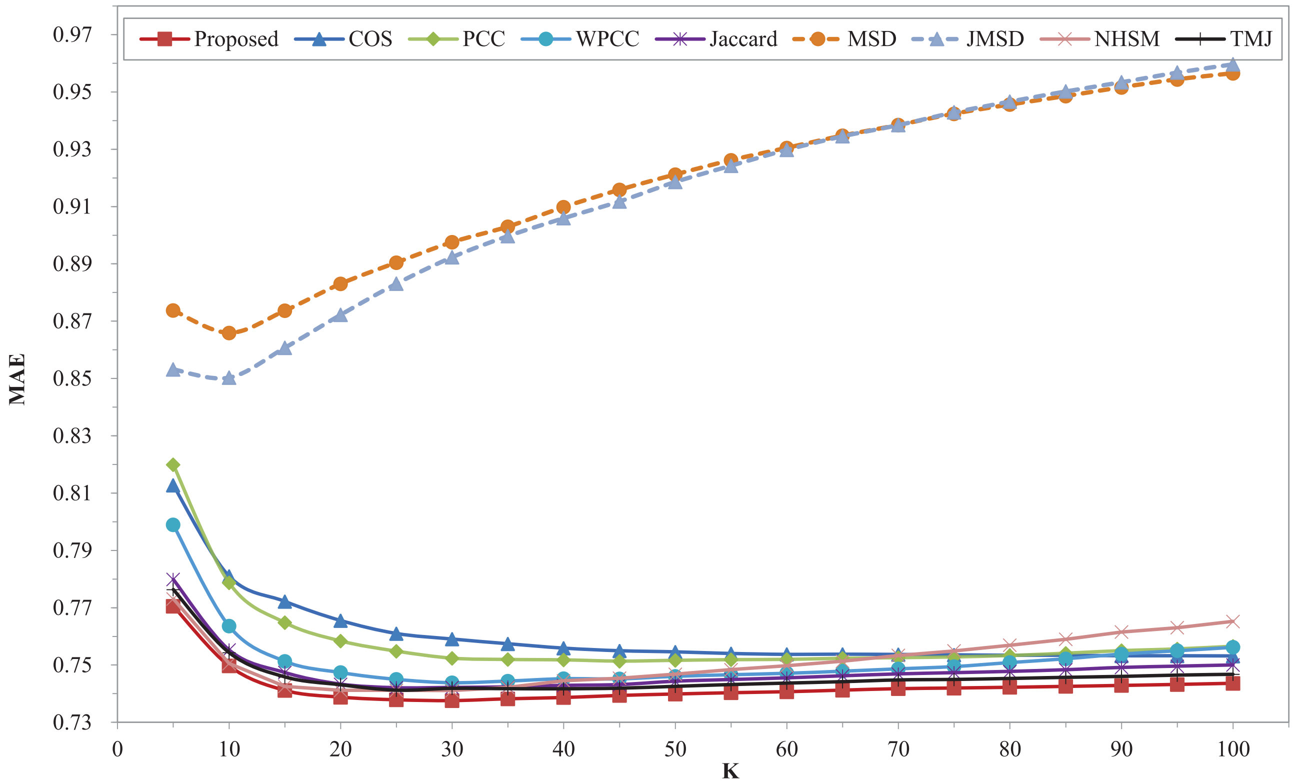

Fig 2. MAE of different similarity measures with different K on Movielens 100K dataset.

https://doi.org/10.1371/journal.pone.0204003.g002

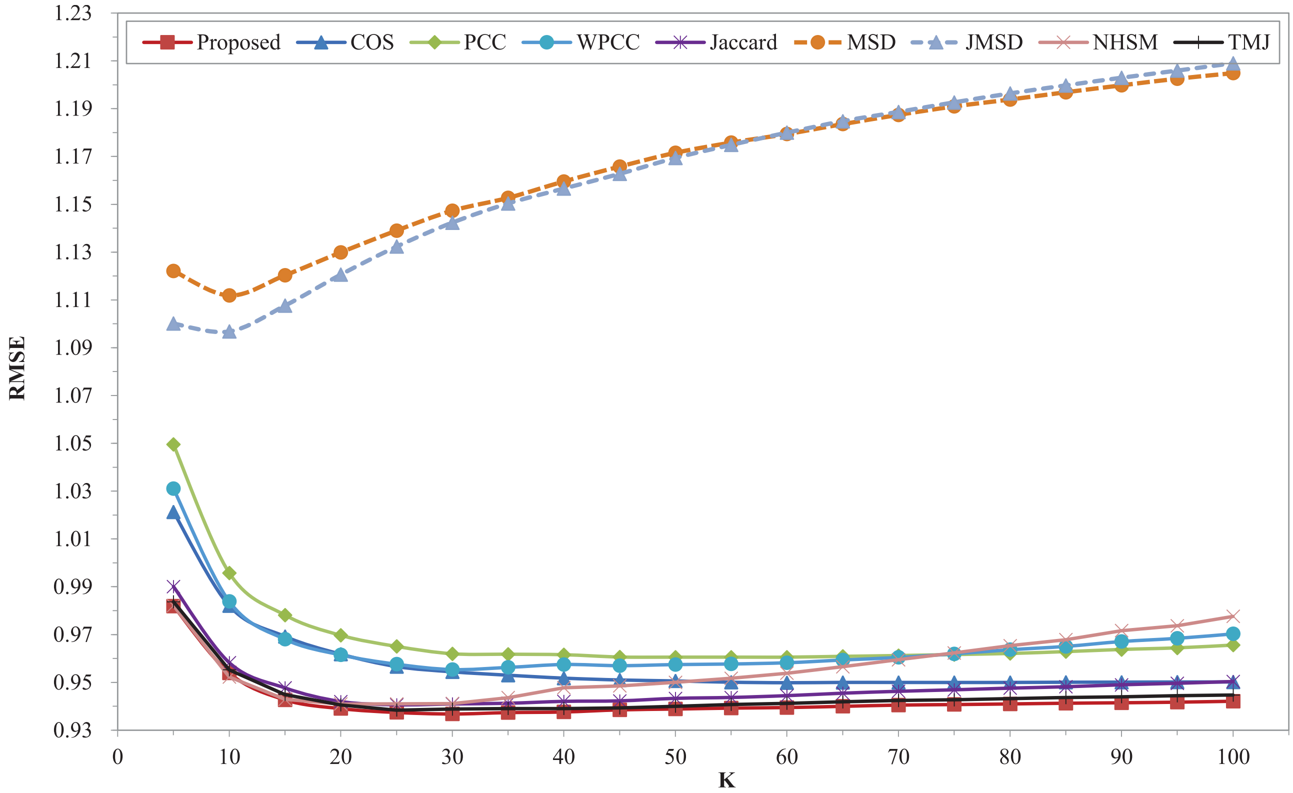

Fig 3. RMSE of different similarity measures with different K on Movielens 100K dataset.

https://doi.org/10.1371/journal.pone.0204003.g003

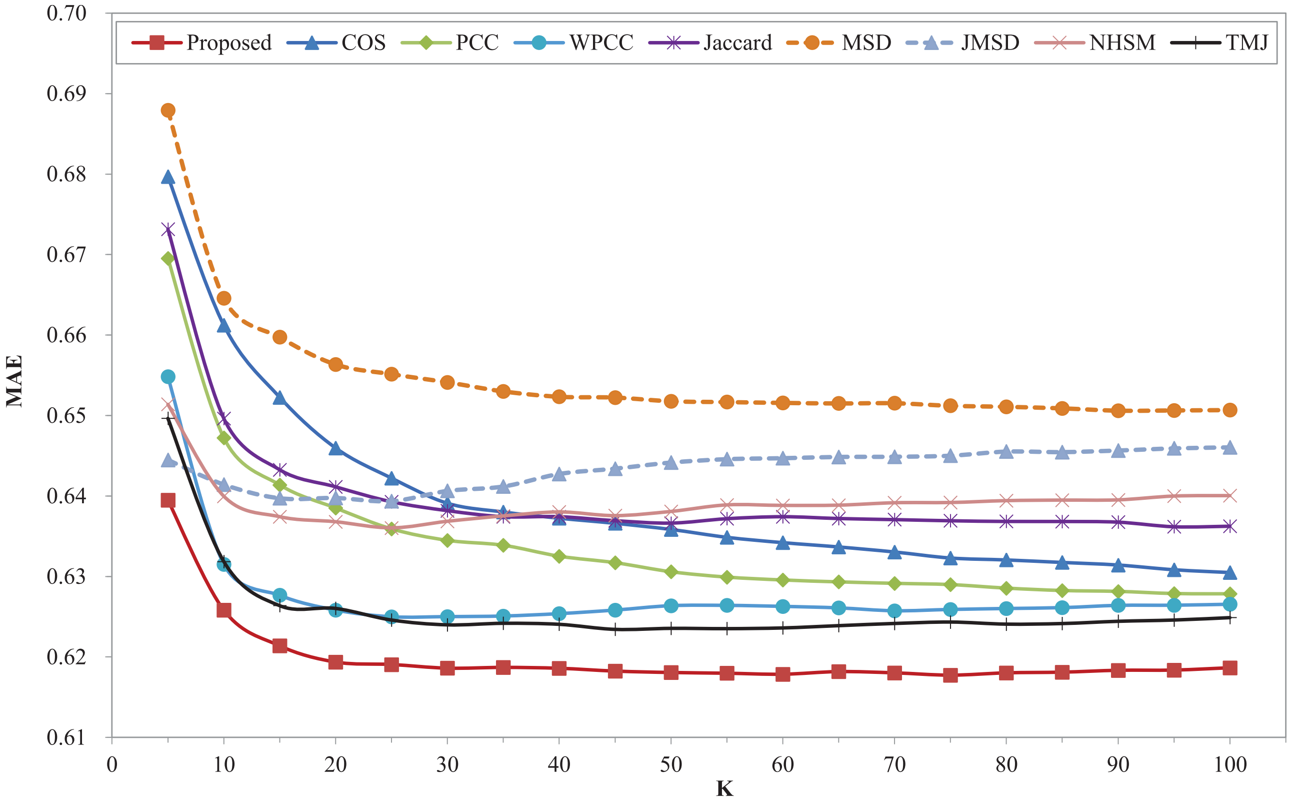

Fig 4. MAE of different similarity measures with different K on FilmTrust dataset.

https://doi.org/10.1371/journal.pone.0204003.g004

COS. The two curves of COS measure decrease within the range of K. The two plots clearly shows that our proposed model surpasses other measures over the entire range of K, while, the two measures of MSD and JMSD are apparently inferior to other similarity measures. Our measure has the best MAE and RMSE when K = 30. The performance of TMJ is closest to our proposed model. It can be observed that the classic measures of COS, PCC, WPCC, MSD and JMSD exhibit larger values of MAE and RMSE in the whole range. The MAE and RMSE of NHSM measure simultaneously reach the lowest point when K is 20, which increases significantly with the increase of K when it is greater than 20. The Jaccard measure owns a good result in this dataset. Moreover, our proposed model is more stable than other measures throughout the nearest neighbors.

The nine different similarity measures are executed on FilmTrust dataset. The prediction errors of MAE and RMSE with different number of nearest neighbors are shown in Figs 4 and 5. Compared with the recently proposed measure of TMJ, the improvement of our proposed model is remarkable. The two figures clearly show that our proposed measure outperforms all other measures, and it has the best results of MAE and RMSE when K = 75. In this dataset, the MSD measure has the worst MAE and RMSE over the entire range of K. The performance of TMJ measure ranks second in the whole range. The classic measures of COS, PCC, WPCC, Jaccard, JMSD and NHSM have worse results in these two metrics. Moreover, the WPCC measure is better than PCC when K is less than 70. Compared with TMJ, our proposed model has at least 1.56% and 0.96% higher improvement in terms of the MAE and RMSE respectively. Compared with COS, our proposed model has at least 7.04% and 5.4% higher improvement in terms of the MAE and RMSE respectively.

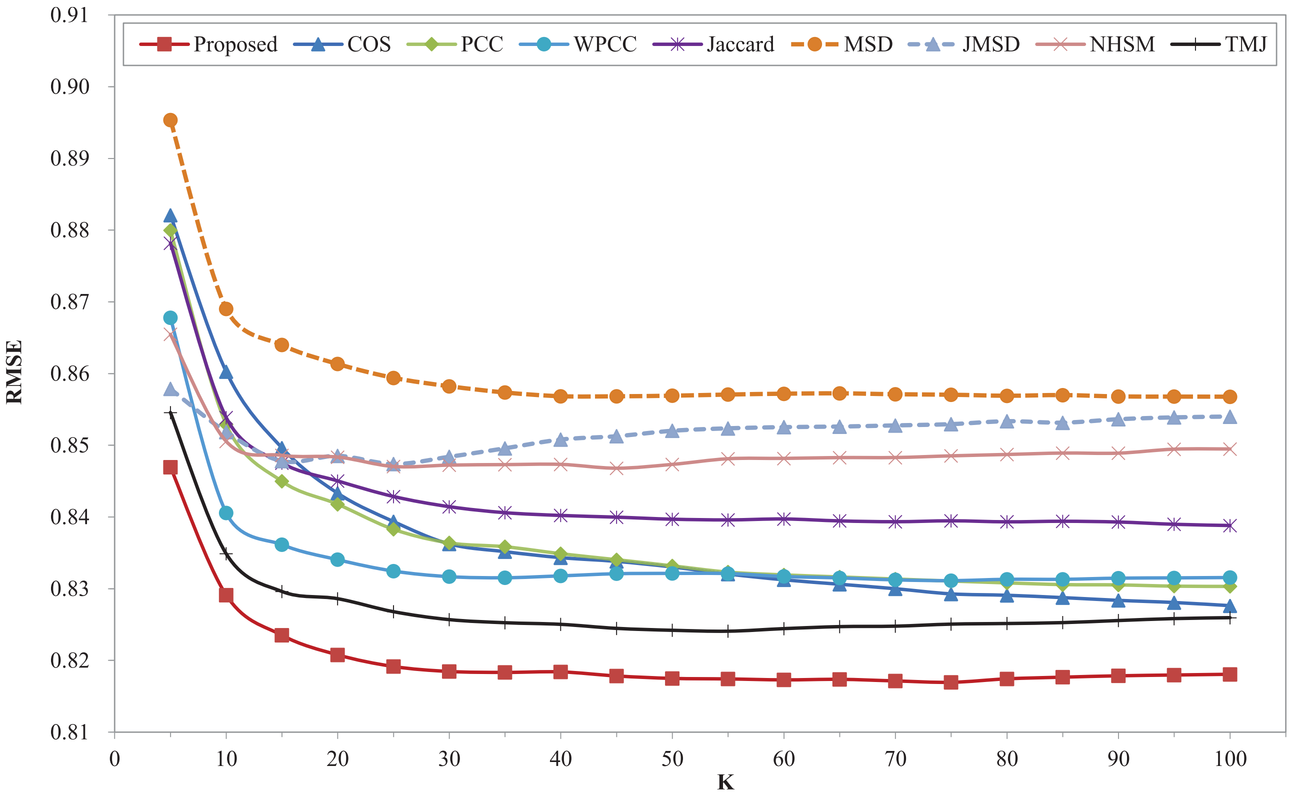

Fig 5. RMSE of different similarity measures with different K on FilmTrust dataset.

https://doi.org/10.1371/journal.pone.0204003.g005

Figs 6 and 7 show the prediction errors of nine similarity measures on the Ciao dataset. It can be seen that our proposed model is apparently superior to other measures over the entire range of K in terms of MAE and RMSE, and the curves of our proposed model descend slowly over the entire K value range. The MAE and RMSE of some classic similarity measures including COS, PCC, WPCC, Jaccard, MSD, JMSD, NHSM are comparatively high. Especially the COS similarity measure, which obtains the worst results in the whole range. The recently proposed similarity measure of TMJ obtains a good result of MAE and RMSE. Compared with TMJ, the MAE and RMSE of our proposed model reduce at least 5.58% and 4.4% respectively. Compared with COS measure, the MAE and RMSE of our proposed model reduce at least 17.54% and 17% respectively. Furthermore, the performance of all measures is relatively stable throughout the nearest neighbors.

Finally, the prediction errors of the nine CF measures are tested on Epinions dataset. The results of MAE and RMSE with different K are shown in Figs 8 and 9. It can be seen that our proposed model obtains best results in the whole range compared with all other measures. The MAE and RMSE of our measure decrease with the increasing of K, while the measures results of COS, PCC, WPCC, Jaccard, MSD, JMSD and NHSM increase with the increasing of K. In this dataset, the NHSM measure owns the worst results in the whole range. The classic measures of COS, PCC, WPCC, Jaccard, MSD and JMSD have worse results in these two metrics. Compared with PCC, the WPCC measure achieves worse results over the entire range of K. The performance of the TMJ measure is stable and ranks second. Compared with TMJ, the MAE and RMSE of our proposed model reduce at least 4.5% and 7.9% respectively. Compared with COS measure, the MAE and RMSE of our proposed model reduce at least 28.6% and 27% respectively.

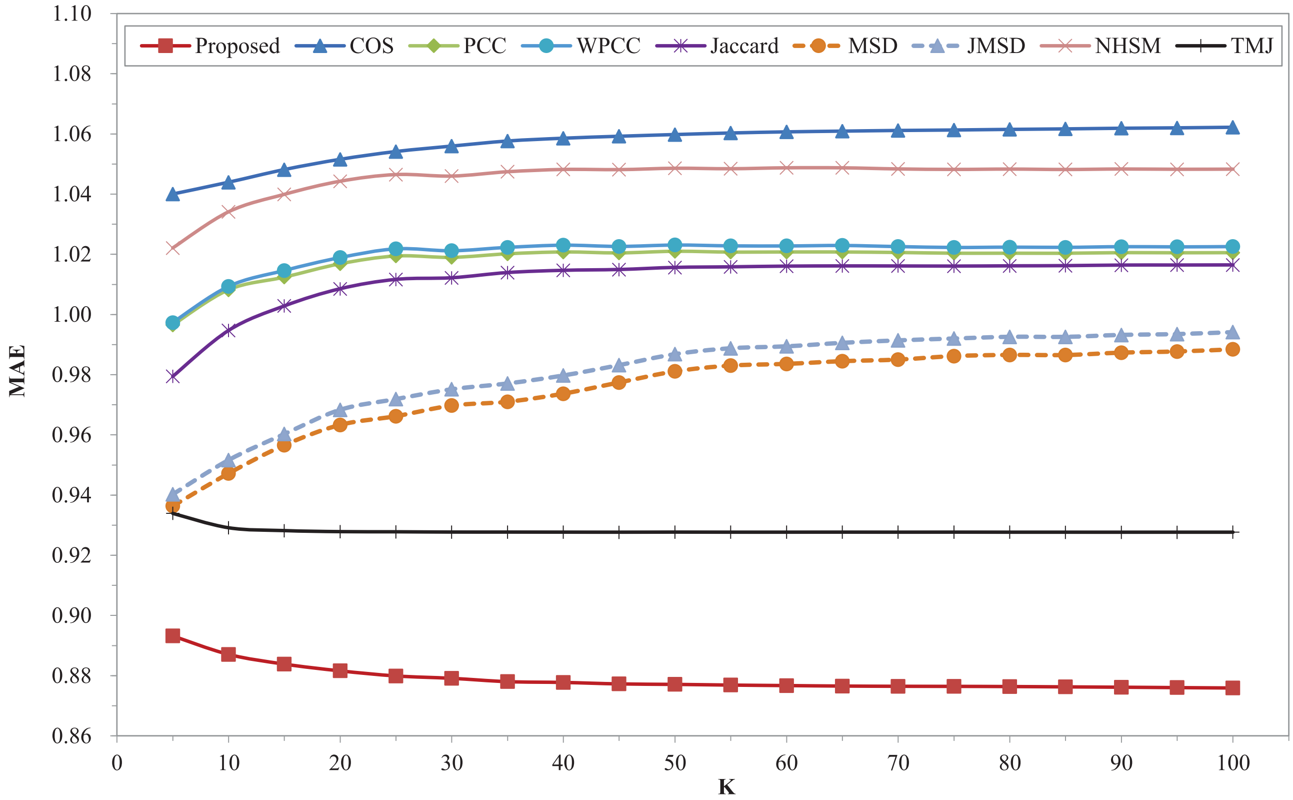

Fig 6. MAE of different similarity measures with different K on Ciao dataset.

https://doi.org/10.1371/journal.pone.0204003.g006

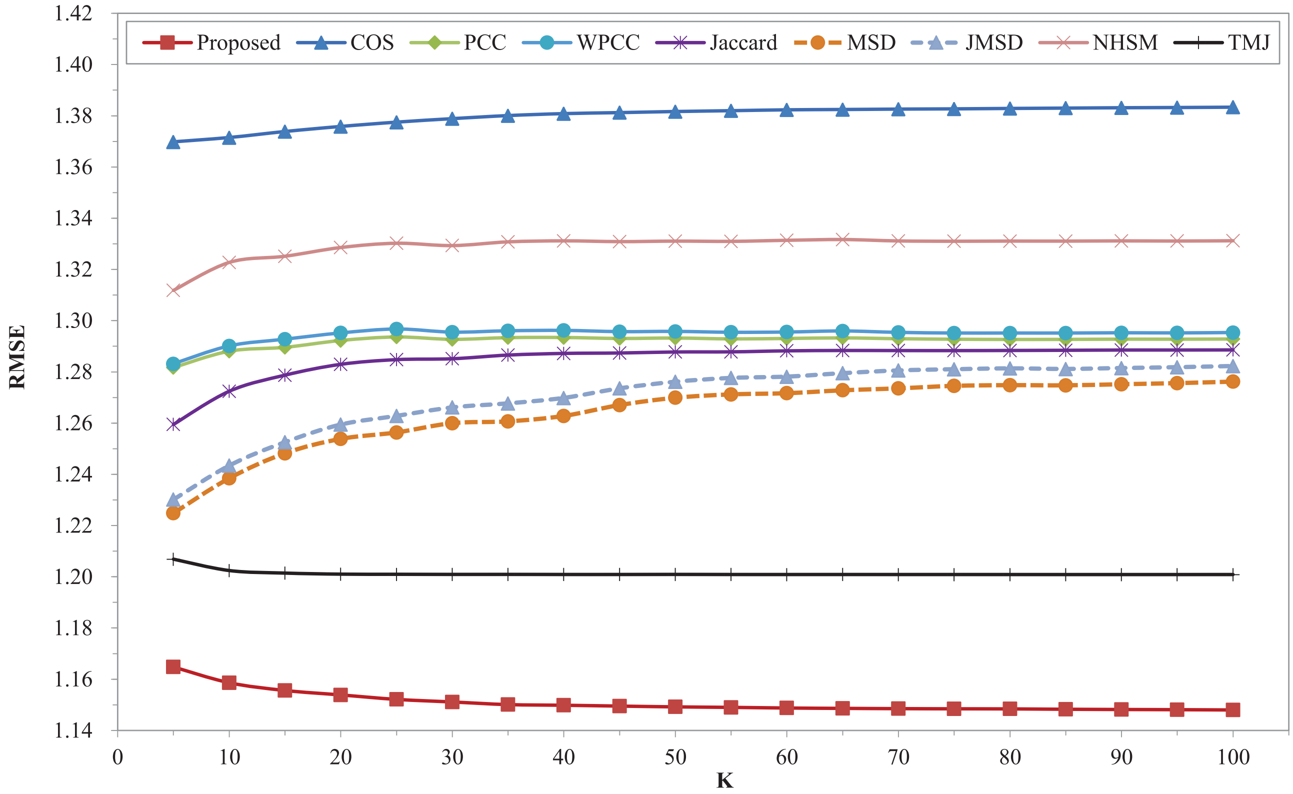

Fig 7. RMSE of different similarity measures with different K on Ciao dataset.

https://doi.org/10.1371/journal.pone.0204003.g007

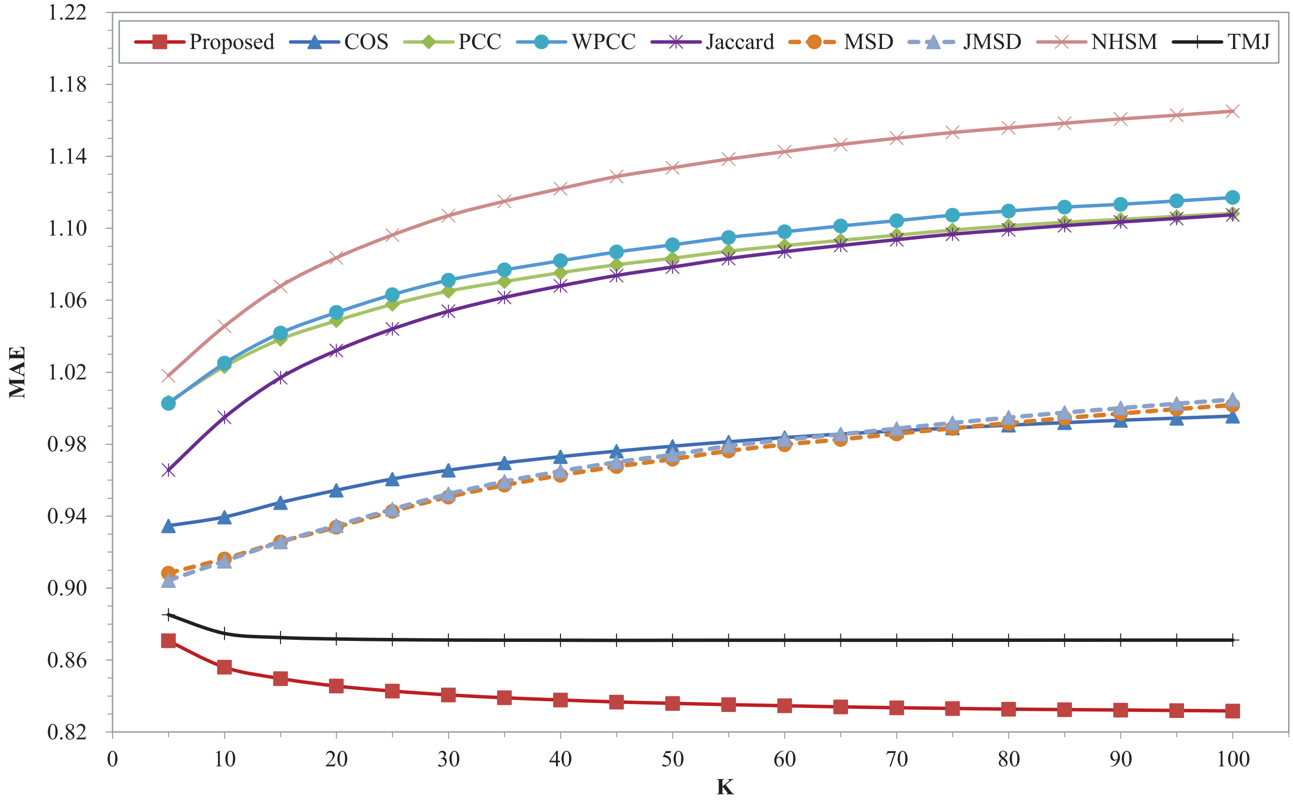

Fig 8. MAE of different similarity measures with different K on Epinions dataset.

https://doi.org/10.1371/journal.pone.0204003.g008

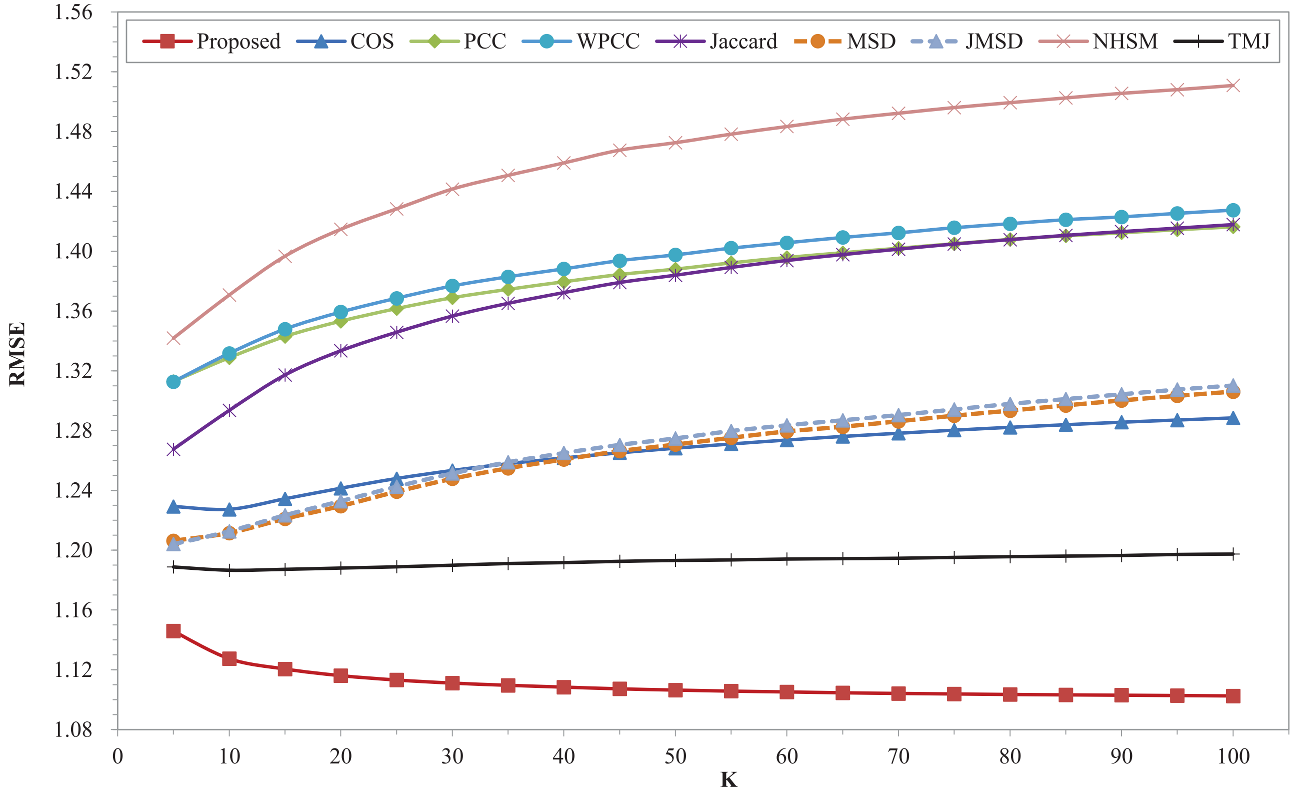

Fig 9. RMSE of different similarity measures with different K on Epinions dataset.

https://doi.org/10.1371/journal.pone.0204003.g009

5 Discussions

In this paper, a novel similarity model is proposed, which is constructed of three impact factors, S_1, S_2 and S_3. S_1 defines the similarity between users. It is divided into two levels based on sparsity of the dataset in the RS to improve accuracy and ensure efficiency. S_2 is used to punish the small proportion of co-rated items. User's rating preference is weighted by S_3.

The experiments are implemented on four datasets with different sparsity levels, and the sparsity of the four training datasets (Movielens 100K, FilmTrust, Ciao, Epinions) is 94.956%, 99.09%, 99.98%, and 99.99% respectively. The sparsity levels are constantly increasing. Our proposed model is compared with other eight measures such as COS, PCC, WPCC, Jaccard, MSD, JMSD, NHSM and TMJ in two metrics of MAE and RMSE. The experiment results show that our measure achieves better performance on four datasets compared with all other measures, especially on extremely sparse datasets. Compared with the recently proposed measure of TMJ, although the sparsity of the Movielens 100K is the lowest, the advantage of our model is not obvious. On other three datasets, our model achieves remarkable improvement. Compared with all other measures, the MAE and RMSE of our proposed model reduce 1.56%—7.04% and 0.96%—5.4% respectively on FilmTrust dataset, the MAE and RMSE of our proposed model reduce at least 5.58% -17.54% and 4.4%—17% respectively on Ciao dataset, and the MAE and RMSE of our proposed model reduce at least 4.5%—28.6% and 7.9%—27% respectively on Epinions dataset. From the data analysis, it is seen that the advantages of our model are more pronounced with the increasing of sparsity. In addition, the performance of our model is relatively stable on all four datasets.

Therefore, the experiment results verify that effectiveness of our proposed similarity model, and it is more suitable for RS especially on extremely sparse recommender systems.

6 Conclusions and future work

In this paper, we concentrate on improving the preference and quality of recommendation in case of sparsity data. To alleviate this problem, an improved similarity model is proposed. Three similarity impact factors were taken into account in the proposed model. To validate the effectiveness of the proposed measure, we employed four popular datasets of Movielens 100K, FilmTrust, Ciao, Epinions. Results suggest that the proposed similarity measure can effectively improve the accuracy of recommendation when the data is sparsity, and it can overcome the drawbacks of the traditional similarity measures.

The academic contribution of our work can be summarized as follows. First, our model make full use of the rating data to improve the accuracy of recommender systems. All the rating data from users is used in the model, not just the co-rated rating data. Second, the problem of co-rated items is solved in our model, which still can obtain an accurate similarity when there is no co-rated items between two users. Third, this paper proposes a new similarity model for collaborative filtering approaches, which shows superior performance than the traditional similarity measures such as COS, PCC, WPCC, Jaccard, MSD, JMSD and NHSM. Finally, most studies that alleviate the data sparse problem have designed more complex models or utilized additional content-based information, which will increase the calculation time. Our purpose is to improve the existing traditional similarity measure just based on available rating data, and the proposed measure can be regarded as a substitute for the traditional measures.

However, the proposed similarity measure still suffers from the complete cold start problem. In our future research issues, inspired by one class of collaborative filtering approaches, we plan to adapt a Matrix Factorization framework to address the new user complete cold start problem and further improve the accuracy of the recommendation.

Acknowledgments

This work is supported by the program for Changjiang Scholars and Innovative Research Team in the University of Ministry of Education of China, number IRT_17R87.

Author Contributions

Methodology: Junmei Feng, Xiaoyi Fengs, Jinye Peng.

Project administration: Xiaoyi Fengs, Jinye Peng.

Validation: Junmei Feng.

Writing – original draft: Junmei Feng.

Writing – review & editing: Junmei Feng, Xiaoyi Fengs, Ning Zhang.

References

1. Borchers A, Herlocker J, Konstan J, Riedl J. Internet Watch: Ganging Up on Information Overload. Computer. 1998 Apr; 31(4):106–108. https://doi.org/10.1109/2.666847
2. Bobadilla J, Ortega F, Hernando A, Gutierrez A. Recommender systems survey. Knowledge-Based Systems. 2013 Jul; 46:109–132. https://doi.org/10.1016/j.knosys.2013.03.012
3. Lu J, Wu D, Mao M, Wang W, Zhang G. Recommender system application developments: a survey. Decision Support Systems. 2015 Jun; 74:12–32. https://doi.org/10.1016/j.dss.2015.03.008
4. Yoshii K, Goto M, Komatani K, Ogata T, Okuno HG. An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model. IEEE Transactions on Audio Speech and Language Processing. 2008 Feb; 16(2):435–447. https://doi.org/10.1109/TASL.2007.911503
5. Oh J, Kim S, Kim J, Yu H. When to recommend: A new issue on TV show recommendation. Information Sciences. 2014 Oct; 280:261–274. https://doi.org/10.1016/j.ins.2014.05.003
6. Palopoli L, Rosaci D, Sarne GML. Introducing Specialization in e-Commerce Recommender Systems. Concurrent Engineering Research and Applications. 2013; 21(3):187–196. https://doi.org/10.1177/1063293X13493915
7. Lee HJ, Park SJ. MONERS: A news recommender for the mobile web. Expert Systems with Applications. 2007 Jan; 32(1):143–150. https://doi.org/10.1016/j.eswa.2005.11.010
8. Hwang RH, Hsueh YL, Chen YT. An effective taxi recommender system based on a spatio-temporal factor analysis model. Information Sciences. 2015 Sep; 314:28–40. https://doi.org/10.1016/j.ins.2015.03.068
9. Gavalas D, Konstantopoulos C, Mastakas K, Pantziou G. Mobile recommender systems in tourism. Journal of Network and Computer Applications. 2014 Mar; 39:319–333. https://doi.org/10.1016/j.jnca.2013.04.006
10. Xia Z, Feng X, Peng J, Fan J. Content-Irrelevant Tag Cleansing via Bi-Layer Clustering and Peer Cooperation. Journal of Signal Processing Systems. 2015 May; 81(1):29–44. https://doi.org/10.1007/s11265-014-0895-y
11. Cacheda F, Carneiro V, Fernandez D, Formoso V. Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. Acm Transactions on the Web. 2011 Feb; 5(1):1–33. https://doi.org/10.1145/1921591.1921593
12. Adomavicius G, Zhang J. Classification, Ranking, and Top-K Stability of Recommendation Algorithms. Informs Journal on Computing. 2016 May; 28(1):129–147. https://doi.org/10.1287/ijoc.2015.0662
13. Al-Shamri MYH. User profiling approaches for demographic recommender systems. Knowledge-Based Systems. 2016 May; 100:175–187. https://doi.org/10.1016/j.knosys.2016.03.006
14. Pazzani MJ, Billsus D. Content-Based Recommendation Systems. Adaptive Wed in the Adaptive Web. 2007 Jan; 4321:325–341. https://doi.org/10.1007/978-3-540-72079-9_10
15. Kim HN, Ha I, Lee KS, Jo GS, El-Saddik A. Collaborative user modeling for enhanced content filtering in recommender systems. Decision Support Systems. 2011 Nov; 51(4):772–781. https://doi.org/10.1016/j.dss.2011.01.012
16. Zhang HR, Min F, Zhang ZH, Wang S. Efficient collaborative filtering recommendations with multi-channel feature vectors. International Journal of Machine Learning and Cybernetics. 2018; 4:1–8.

17. Wang HC, Jhou HT, Tsai YS. Adapting Topic Map and Social Influence to the Personalized Hybrid Recommender System. Information Sciences. 2018 Apr; 000:1–17.

18. Linden G, Smith B, York J. Amazon.com recommendations: item-to-item collaborative filtering. IEEE Internet Computing. 2003 Jan; 7(1):76–80. https://doi.org/10.1109/MIC.2003.1167344

19. Wang Y, Deng J, Gao J, Zhang P. A Hybrid User Similarity Model for Collaborative Filtering. Information Sciences. 2017 Dec; 418-419:102–118. https://doi.org/10.1016/j.ins.2017.08.008

20. Liu H, Hu Z, Mian A, Tian H, Zhu X. A new user similarity model to improve the accuracy of collaborative filtering. Knowledge-Based Systems. 2014 Jan; 56:156–166. https://doi.org/10.1016/j.knosys.2013.11.006

21. Ren L, Wang W. An SVM-based collaborative filtering approach for Top-N web services recommendation. Future Generation Computer Systems. 2018 Jan; 78:531–543. https://doi.org/10.1016/j.future.2017.07.027

22. Hernando A, Ortega F. A probabilistic model for recommending to new cold-start non-registered users. Information Sciences. 2017 Jan; 376:216–232. https://doi.org/10.1016/j.ins.2016.10.009

23. Hu QY, Zhao ZL, Wang CD, Lai JH. An Item Orientated Recommendation Algorithm from the Multi-view Perspective. Neurocomputing. 2017 Dec; 269:261–272. https://doi.org/10.1016/j.neucom.2016.12.102

24. Feng X, Wu S, Tang Z, Li Z. Sparse latent model with dual graph regularization for collaborative filtering. Neurocomputing. 2018 Apr; 284:128–137. https://doi.org/10.1016/j.neucom.2018.01.011

25. Adomavicius G, Tuzhilin A. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering. 2005 Jun; 17(6):734–749. https://doi.org/10.1109/TKDE.2005.99

26. Ahn HJ. A new similarity measure for collaborative filtering to alleviate the new user cold-starting proble. Information Sciences. 2008 Jan; 78:37–51. https://doi.org/10.1016/j.ins.2007.07.024

27. Shi Y, Larson M, Hanjalic A. Collaborative Filtering beyond the User-Item Matrix: A Survey of the State of the Art and Future Challenges. Acm Computing Surveys. 2014 Apr; 47(1):1–45. https://doi.org/10.1145/2556270

28. Herlocker JL, Konstan JA, Borchers A, Riedl J. An algorithmic framework for performing collaborative filtering. In Proceedings of the 23rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Athens: ACM, 1999, pp.230–237.

29. Kim HN, El-Saddik A, Jo GS. Collaborative error-reflected models for cold start recommender systems. Decision Support Systems 2011. Jun; 51(3):519–531. https://doi.org/10.1016/j.dss.2011.02.015

30. Jamali M, Ester M. TrustWalker: a random walk model for combining trustbased and item-based recommendation. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York: ACM, 2009, pp.397–406.

31. Koutrika G, Bercovitz B, Garcia-Molina H. FlexRecs: expressing and combining flexible recommendations. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, New York: ACM, 2009, pp.745–758.

32. Bobadilla J, Serradilla F, Bernal J. A new collaborative filtering metric that improves the behavior of recommender systems. Knowledge.-Based Systems. 2010 Aug; 23(6):520–528. https://doi.org/10.1016/j.knosys.2010.03.009

33. Bobadilla J, Serradilla F, Bernal J. A collaborative filtering approach to mitigate the new user cold start problem. Knowledge-Based Systems. 2012 Feb; 26:2252–38. https://doi.org/10.1016/j.knosys.2011.07.021

34. Polatidis N, Georgiadis CK. A muti-level collaborative filtering method that improves recommendations. Expert Systems With Applications. 2016; 48:100–110. https://doi.org/10.1016/j.eswa.2015.11.023

35. Patra BK, Launonen R, Ollikainen V, Nandi S. A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data. Knowledge-Based Systems. 2015 Jul; 82:163–177. https://doi.org/10.1016/j.knosys.2015.03.001

36. Zhang F, Gong T, Lee VE, Zhao G, Rong C, Qu G. Fast algorithms to evaluate collaborative filtering recommender systems. Knowledge-Based Systems. 2016 Mar; 96(C):96–103.

37. Lee J, Lee D, Lee YC, Hwang WS, Kim SW. Improving the accuracy of top- N, recommendation using a preference model. Information Sciences. 2016 Jun; 348:290–304. https://doi.org/10.1016/j.ins.2016.02.005

38. Sun SB, Zhang ZH, Dong XL, Zhang HR, Li TJ, Zhang L, et al. Integrating Triangle and Jaccard similarities for recommendation. Plos One. 2017 Aug; 12(8):e0183570. https://doi.org/10.1371/journal.pone.0183570 PMID: 28817692

39. Polato M, Aiolli F. Exploiting Sparsity to Build Efficient Kernel Based Collaborative Filtering for Top-N Item Recommendation. Neurocomputing. 2017 Dec; 268:17–26. https://doi.org/10.1016/j.neucom.2016.12.090

40. Camacho LAG, Alves-Souza SN. Social network data to alleviate cold-start in recommender system: A systematic review. Information Processing and Management. 2018 Jul; 54(4):529–544. https://doi.org/10.1016/j.ipm.2018.03.004

41. Cheng K, Wang L, Shen Y, Wang H, Wang Y, Jiang X, et al. Secure k-nn query on encrypted cloud data with multiple keys. IEEE Transactions on Big Data. 2017; 1–14. https://doi.org/10.1109/TBDATA.2017.2707552

42. Peng M, Zeng G, Sun Z, Huang J, Wang H, Tian G. Personalized app recommendation based on app permissions. World Wide Web-internet and Web Information Systems. 2017; 21(8):1–16.

43. Herlocker JL, Konstan JA, Terveen LG, Riedl JT. Evaluating collaborative filtering recommender systems. Acm Transactions on Information Systems. 2004 Jan; 22(1):5–53. https://doi.org/10.1145/963770.963772