Modern Management based on Big Data III
A.J. Tallón-Ballesteros (Ed.)

doi:10.3233/FAIA220091

Collaborative Filtering Algorithm Based on Item Popularity and Dynamic Changes of Interest

Xuan Zhang[a,b],Kai Su[a,1],Feng Qian[a],Yanru Zhang[c],Kan Zhang[a]
[a]*Department of Management Engineering and Equipment Economics, Naval University of Engineering,Wuhan 430033, China*
[b]*Naval Aeronautical University, Yantai 264000, China*
[c]*Bejing Goldwind New Energy Trade Co.LTD, Beijing 102600, China*

Abstract. The traditional collaborative filtering algorithm does not consider the influence of item popularity in similarity calculation, and the prediction score does not consider the influence of time on the change of user interest, resulting in inaccurate similarity calculation and single recommendation result. To solve these problems, this paper improved the traditional similarity calculation method by combining the item popularity penalty coefficient, improved the recommendation diversity of the algorithm, and integrated the time factor into the prediction method to solve the problem of interest attenuation. Experiments on the 100K and 1M data set of Movielens show that the improved algorithm effectively improves the accuracy and coverage of recommendations.

Keywords. Recommendation system; Collaborative filtering; Item popularity; Dynamic interest change; Time function.

1. Introduction

Recently, Customers cannot quickly find satisfactory products in the face of excessive information, resulting in poor online shopping experience for users [1]. Recommendation system can explore potential interests and hobbies by analyzing the historical behavior information of users and make targeted personalized recommendations to users without specific the needs of users [2].

Collaborative filtering is considered to be one of the most promising and widely used recommendation algorithms, which is used to help users finding commodities they may like [3, 4]. Traditional collaborative filtering methods calculate similarity only based on user grade [5]. However, with the increase of the number of users and commodities, users are more easily to find and buy those popular commodities, causing the asymmetry of score data. In this case, the similarity calculated is not accurate, and the popular items in the generated recommendation list almost account for the majority, which is not conducive to the individuation and novelty of the recommendation [6]. This situation is known as the "long tail effect" in the recommendation system [7]. In

[1]Corresponding author: Kai Su, Naval University of Engineering, Wuhan 430033, China. Email: keppelsue@163.com. This research work is supported by the National Natural Science Foundation of China (NSFC) under Grant No. 61802425 and National Social Science Foundation of China under 19CGL073 and 2021-SKJJ-C-017.

addition, the user needs and interests will change at different stage. The traditional methods calculation methods for all score right, unable to identify the dynamic changes of the user's interests which called “drift” [8].

In this paper, a collaborative filtering algorithm based on item popularity and dynamic change of interest is proposed. Firstly, the item popularity was integrated into the similarity calculation method, and the popularity penalty function was defined through item popularity and item popularity differences to improve the diversity of recommendation results. Secondly, according to the different behavior characteristics of users, the time decay function is defined to weaken the contribution of old data to the predicting results and make the final recommendation result more accurate. The experimental results on Movielens data set show that the proposed algorithm can not only improve the recommendation accuracy, but also improve the diversity of recommendations.

2. Related work

In recent years, collaborative filtering based recommendation algorithms have been widely used to solve personalized recommendation in the field of e-commerce, among which computing similarity and predicting score are the most important two parts. Item-based Collaborative Filtering algorithm (IBCF) calculates the similarity between items according to users' scores, and firstly constructs a scoring matrix based on users' scoring information. Based on the constructed scoring matrix, similarity calculation method was used to calculate the similarity between items [9]. The commonly used similarity calculation methods include Pearson similarity and modified cosine similarity [10]. The algorithm in this paper takes modified cosine similarity as the similarity calculation method, as shown in Equation (1). Then, the top-N items with the highest similarity of target items are taken as neighbors, and the prediction score of users on target items is obtained by using the prediction formula, as shown in Formula (2).

$$Sim(i,j)=\frac{\sum_{u\in U_{ij}}(r_{ui}-\overline{r}_i)(r_{uj}-\overline{r}_j)}{\sqrt{\sum_{u\in U_i}(r_{ui}-\overline{r}_i)^2}\sqrt{\sum_{u\in U_j}(r_{uj}-\overline{r}_j)^2}} \tag{1}$$

$$P_{ui}=\overline{r}_i+\frac{\sum_{j\in N}sim(i,j)\times(r_{uj}-\overline{r}_j)}{\sum_{j\in N}|sim(i,j)|} \tag{2}$$

In order to alleviate the “long tail effect” of recommendation system, scholars introduced popularity penalty factor into the algorithm. Gao et al. [11] proposed a method to punish popular items. They took the number of popular items and the proportion of the total items as punishment, and added it into the similarity calculation method. Hao et al. [12] introduced item popularity as a weight factor into similarity calculation and recommendation process to improve the reliability of user similarity calculation and influence of unpopular items in the final item recommendation process. Wei et al. [13] proposed a collaborative filtering recommendation algorithm combined with item popularity weighting, analyzed the influence of item popularity and popularity differences on similarity, and designed penalty weights for popular items exceeding the popularity threshold to reduce their contribution in similarity calculation (IPCF).

The introduction of popularity penalty weight improves the algorithm's ability to mine unpopular items, but it cannot dynamically recommend items to users. In order to solve the problem of interest dynamic change, the time factor is integrated into the algorithm. Yi [14] put forward a kind of computing time weighting algorithm to track each user's interest changes. By introducing personalized attenuation factor, the algorithm makes each score more reasonable and effective. However, the purchase cycle of each user is different, so it is difficult to provide personalized recommendation for different users. Chen [15] et al proposed a recommendation method based on dynamic time attenuation (TWCF). TWCF dynamically determines the attenuation function based on users' scoring behavior, gradually weakens the influence of old data and accurately predicts future users' preferences. Song et al. [16] proposed the time-weighted based information recommendation algorithm, where the users set, time, tag set and goods resources are utilized to calculate the tag feature vector to predict the user's preferences. After time function is added into the recommendation algorithm, the problem of dynamic change of user interest is solved to some extent.

3. Collaborative filtering algorithm based on item popularity

3.1. Normalization of item popularity

In the recommender system, item popularity is expressed as the number of user evaluations. The more times an item is evaluated, the higher its popularity will be. Popular items are more likely to be selected and evaluated by users due to their popularity or high cost performance, and two popular items are more likely to be scored by the same user at the same time. When using traditional similarity to calculate the similarity of two popular items, the calculated similarity is higher, but this does not mean that popular items are similar to other items.

Due to the large difference in popularity between items, there will be a large difference in numerical value in calculation, and the result will be greater than 1 in the subsequent calculation of attribute weight function. Therefore, this paper normalized The Times of user evaluation, as shown in equation (3), to keep its value range at [0,1], so as to reduce data deviation. Where, popitem (i) refers to the number of times that item i has been evaluated, popmax refers to the number of times that the most popular item has been evaluated, and popmin refers to the number of times that the least popular item has been evaluated.

$$Pop(i) = \frac{popitem(i) - popmin}{popmax - popmin} \tag{3}$$

3.2. Deviation of item popularity

In this paper, the absolute value of the difference in item popularity is defined as the difference in item popularity. The smaller the difference in popularity between the two items, the closer the popularity of the two items. Items with small differences in popularity are similar in popularity and other aspects. Such items are more likely to be discovered and purchased by the same user. Therefore, there are more users who have jointly evaluated these two items, and the calculated similarity will be high. The difference in prevalence between item I and J is shown as follows:

$$popBias(i,j) = | Pop(i) - Pop(j) | \tag{4}$$

3.3. Weight of popularity

Items with high popularity have more common scores with other items, popular items are easier to be selected and evaluated by users, their similarity with other items is generally high, and such items are subject to greater punishment. Therefore, the popularity of items is positively correlated with the weight of punishment. Popularity differences will also affect the calculation of similarity. The smaller the difference in popularity between the two items, the greater the possibility of them being found and purchased by user, causing the similarity being evaluated too high. Based on the above analysis, combined with the popularity of the item and the differences in popularity, the popularity penalty weight function is proposed:

$$Weight(i,j) = \frac{lg2 \times Pop_i}{lg(2 + Popbais_{i,j})} \tag{5}$$

Pop (i) is the normalized popularity of item i, and PopBais (i, j) is the difference in popularity between item i and item j. Since the prevalence difference between items is large, the numerical value will have a great influence on the penalty weight, so lg function is introduced to reduce the numerical influence of the prevalence difference. When PopBais (i, j) = 0, the prevalence difference is the smallest. Combining the popularity penalty weight function with the traditional similarity calculation method, the improved item similarity calculation formula is as follows:

$$Sim(i,j) = \frac{\sum_{u \in U_{ij}} [(r_{ui} - \overline{r_i}) weight(i)][(r_{uj} - \overline{r_j}) weight(j)]}{\sqrt{\sum_{u \in U_i} [(r_{ui} - \overline{r_i}) weight(i)]^2} \sqrt{\sum_{u \in U_j} [(r_{uj} - \overline{r_j}) weight(j)]^2}} \tag{6}$$

4. Collaborative filtering algorithm based on dynamic temporal interest change

Users' purchase interest and memory both change over time, and it is difficult for users to maintain long-term interest in a product. Generally speaking, it shows a declining trend. The recent purchase data should have a greater contribution to the prediction of preferences, so this paper gives different weights to users for each score according to the time of prediction, so as to weaken the contribution of old data to the prediction of scores. The user's interest changes dynamically with time, with more emphasis on the recent purchase interest. The characteristic of the exponential function is that it attenuates sharply first and then slows down. Therefore, this paper chooses exponential function as the attenuation function of user interest. A decay coefficient ε is introduced to slow down the decay rate. Let ε=1/T, then the time function is shown as follows.

$$f(t) = e^{-\varepsilon t} = e^{-\frac{t}{T}} \tag{7}$$

T is the time period, and T is inversely proportional to the ε attenuation coefficient. FIG. 1 shows the curve of time function f under different T values.

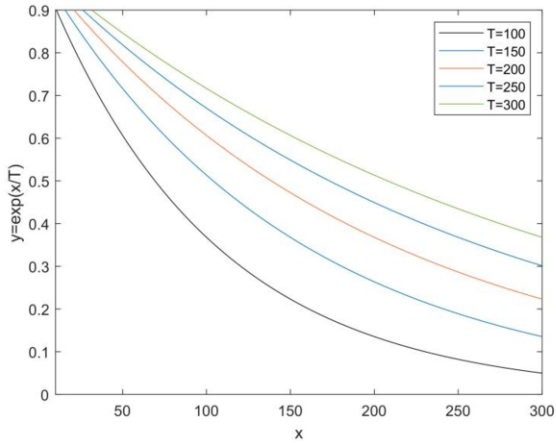


Fig. 1. Curves of f time function with different values of T

The larger T is, the slower the change of time function is and the smaller the contribution of historical data is. The value of T determines the attenuation rate of f to historical data, and an appropriate parameter T should be selected to accurately predict users' future preferences so as to improve the performance of the algorithm. In the recommender system, the decay rate of historical data should be determined by the purchase behavior of each user. In a period of time, if a user has a long and frequent purchase cycle, all the goods purchased in the purchase cycle can provide more accurate data support for predicting preferences. The old data should be decayed slowly and a higher value T should be assigned to the time function. In this case, different T values can be selected according to the shopping cycle of each user, and the purchase cycle of user U can be defined as $T_u = t_{max} - t_{min}$. At this time, the time function is shown in Formula (8), where T_{MAX} is the earliest time when users purchase goods within a period of time, and T_{MAX} is the latest time when users purchase goods within this period. T is the time when the user buys the target item.

$$f(t)=e^{-\frac{t-T_{MIN}}{T_{MAX}-T_{MIN}}} \tag{8}$$

In addition, the buying habits of users vary, and even the same users have different attitudes towards different things. Therefore, this paper classifies items into instantaneous interest items, general interest items and long-term interest items according to users' purchasing behavior. If the user's interest in a certain item lasts for a long time, it can be considered that the user has a long-term interest in this item. If there is only one purchase record, it can be considered that the user has only transient interest in such items. Anything in between is called general interest. The k-means clustering method was used to cluster commodities according to user rating data. The purchasing frequency α was defined as the rating times of each commodity category by users, and the frequency threshold θ was set. The purchase frequency greater than the threshold θ is classified as the long-term interest of users, while the purchase frequency less than the threshold is classified as the short-term interest of users. The liking degree of long-term interest products does not decrease with time in the interest cycle of users, so the time function has no weakening effect on the score of such products. However, for commodities of general interest and instantaneous interest, users' interest in them

will gradually weaken over time, so the contribution of their scores should be weakened when predicting user preferences. On this basis, redefine the time function:

$$f(t)=\begin{cases}1 & if\ \alpha>\theta \\ e^{-\frac{t-T_{MIN}}{T_{MAX}-T_{MIN}}} & if\ \alpha<=\theta\end{cases} \tag{9}$$

It can be seen from Equation (9) that, for commodities purchased more frequently by users, the time function does not reduce the contribution of scoring, while for commodities purchased less frequently, the time function reduces its contribution according to the purchasing cycle and scoring time of users when predicting scoring. Combined with the time function, the prediction formula is as follows:

$$P_{ui}=\overline{r}_i+\frac{\sum_{j\in N} sim(i,j)\times[(r_{uj}-\overline{r}_j)\times f(t)]}{\sum_{j\in N}|sim(i,j)|} \tag{10}$$

5. Experimental Analysis

5.1. Experimental data set

In order to verify the effectiveness of the algorithm, several experiments are carried out on Movielens 100K and 1M data sets including the ratings of 1682 movies by 943 users. The ratings are divided into five grades from 1 to 5, and each score has a definite scoring time. In this paper, the five-fold crossover method is adopted to divide the data set. The 100K and 1M data sets are randomly divided into five parts, four of which are randomly selected as training sets, and the remaining one as test set. Five data sets are divided into 1-5 respectively, and the average value is taken as the experimental result.

5.2. Evaluation indicators

In this paper, accuracy and coverage are taken as the evaluation indexes of the algorithm. Accuracy is measured by the difference between the predicted score value and the real score, as shown in Equation (11). Coverage rate is a measure of the proportion of recommended items in the total collection of items in the recommendation system, which can effectively reflect the diversity and novelty of recommendations. The formula is shown in Equation (12).

$$MAE=\frac{1}{N}\sum_{i=1}^{m}|p_i-q_i| \tag{11}$$

$$coverage=\frac{\left|\sum_{u\in U} R(u)\right|}{|I|} \tag{12}$$

5.3. Comparison algorithm

Table 1 lists the algorithms used for experimental comparison, including the traditional item-based collaborative filtering algorithm, the recommendation algorithm combined with item popularity, the time-fused collaborative filtering algorithm and the algorithm proposed by this paper.

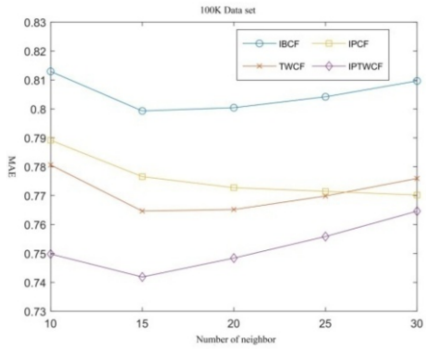
Each algorithm was tested on Movielen-100K and 1M data sets, and MAE and coverage were compared under different numbers of neighbors.

Table 1. Comparison algorithm description

The algorithm name	Item popularity	Time	Algorithm description
Traditional collaborative filtering algorithm(IBCF)			Unimproved item-based collaborative filtering algorithm
Time - based collaborative filtering algorithm(TWCF)[15]		√	Collaborative filtering algorithm combined with time optimization prediction method
Collaborative filtering algorithm based on item popularity(IPCF)[13]	√		A collaborative filtering algorithm was proposed to improve the similarity calculation method based on item popularity
Collaborative filtering algorithm based on dynamic changes of item popularity and interest(IPTWCF)	√	√	We propose a collaborative filtering algorithm combining popularity and time optimization

(1) MAE comparison

Figure 2 and 3 show the MAE comparison results. Obviously, the MAE of the algorithm proposed by this paper is the lowest with different number of neighbors, where the MAE of the traditional collaborative filtering algorithm is the highest. The time-based collaborative filtering algorithm and the popularity-based collaborative filtering algorithm start from the dynamic changes of item popularity and user interest respectively, and combine the popularity penalty weight and time function to optimize the similarity and score prediction respectively. In 100K and 1M data sets with different neighbor numbers, the MAE values of the two algorithms have little difference, and are generally lower than the traditional algorithm. At the same time, this paper improved the algorithm by combining the item popularity and user interest changes, redefined the popularity penalty function and time function, optimized the similarity calculation method and prediction method, and improved the recommendation accuracy. Compared with the MAE based on popularity-based collaborative filtering and time-based collaborative filtering, the recommendation accuracy of the proposed algorithm is higher. Therefore, by adding the time function and popularity penalty function defined in this paper, the similarity between items is more reasonable and preferences can be reasonably predicted according to the dynamic changes of user interests.



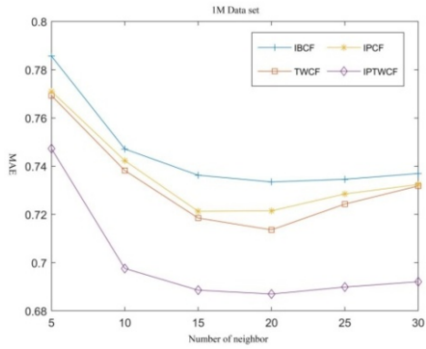
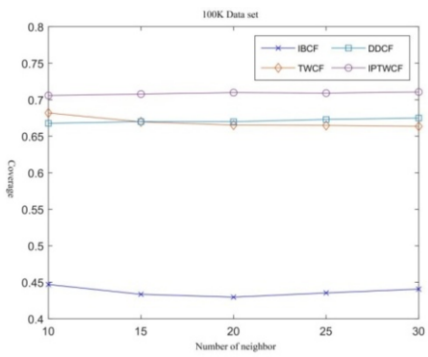


Fig. 2. MAE of different algorithms on Data set 100K **Fig. 3.** MAE of different algorithms on Data set 1M

(2) Comparison of coverage



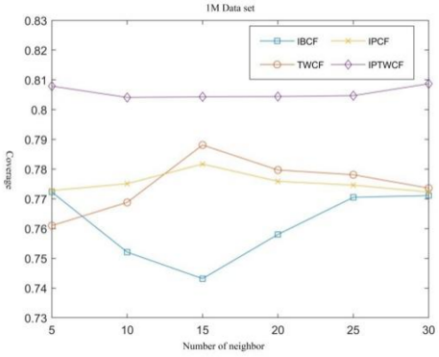


Fig.4. Different algorithms Coverage of Data set 100K **Fig. 5.** Different algorithms Coverage of Data set 1M

Figure 4 and 5 show the coverage comparison results. As can be seen from the comparison test results of the coverage of the two data sets, the coverage of the traditional algorithm is generally low. This is because the items with high popularity are generally similar to other items, and the popular items recommended to users account for the majority, while the number of popular items only accounts for a small part of the total. Therefore, the range of items recommended by the traditional collaborative filtering algorithm is relatively narrow. The collaborative filtering algorithm based on item popularity assigns penalty weight to items with different popularity when calculating similarity, which weakens the influence of popularity on item recommendation. The results show that the coverage of collaborative filtering algorithm based on item popularity is higher than that of traditional collaborative filtering algorithm and time-based collaborative filtering algorithm, indicating that the improvement of popularity can effectively improve the coverage of recommendations. In this paper, based on the collaborative filtering algorithm of popularity and time, the popularity penalty weight is optimized and the recommendation weight of popular items is weakened, thus the potential interests of users are truly explored. Unpopular items can also be recommended to users. As can be seen from the figure 4 and 5, the coverage rate of the algorithm in this paper is the highest under the number of neighbors, indicating that the algorithm in this paper can effectively improve the coverage rate of recommendations and effectively mine the potential interests of users.

6. Conclusions

This paper proposes a collaborative filtering algorithm based on the dynamic change of item popularity and interest, which integrates the item popularity into the similarity calculation method to solve the problem of high similarity of popular item. At the same time, according to the behavior characteristics of each user, a time function is added to the prediction formula to reduce the contribution of historical data to prediction preference. While improving the accuracy of recommendation, the proposed algorithm can effectively mine and recommend the unpopular items in the data set, improve the coverage of recommendation, alleviate the problem of "long tail effect" in the recommendation system, and improve the quality of recommendation.

However, there are also many areas to be improved. In the future work, we will further study the impact of different activity and scoring habits on users' interests, and provide personalized recommendations according to different users' living habits and shopping characteristics.

References

[1] Bresler G, Karz and M.Regret Bounds and Regimes of Optimality for User-User and Item-Item Collaborative Filtering[J]. IEEE Transactions on Information Theory, 2021, PP(99): 1-1.
[2] Meng D F , Liu N , Li M X , et al. An Improved Dynamic Collaborative Filtering Algorithm Based on LDA[J]. IEEE Access, 2021, PP(99):1-1.
[3] Kim S, H Kim, Min J K. An efficient parallel similarity matrix construction on MapReduce for collaborative filtering[J]. Journal of Supercomputing, 2019, 75(1): 123-141.
[4] Alhijawi B, Al-Naymat G, Obeid N, et al. Novel predictive model to improve the accuracy of collaborative filtering recommender systems[J]. Information Systems, 2021, 96: 101670.
[5] Yang E, Huang Y, Liang F, et al. FCMF : Federated collective matrix factorization for heterogeneous collaborative filtering[J]. Knowledge-Based Systems, 2021, 220(1/2): 106946.
[6] Lee Y C, Son J, Kim T, et al. Exploiting uninteresting items for effective graph-based one-class collaborative filtering[J]. The Journal of Supercomputing, 2021, 77(7): 6832-6851.
[7] Manochandar S, Punniyamoorthy M. A new user similarity measure in a new prediction model for collaborative filtering[J]. Applied Intelligence, 2020.
[8] Xiao T, Shen H. Neural variational matrix factorization for collaborative filtering in recommendation systems[J]. Applied Intelligence, 2019.
[9] X Wang, Wang R, Li D, et al. QCF: Quantum Collaborative Filtering Recommendation Algorithm[J]. International Journal of The oretical Physics, 2019.
[10] Xiong H., Chen J., Liu Q., et al. Enhancing Collaborative Filtering by User Interest Expansion via Personalized Ranking[J]. IEEE transactions on systems, man, and cybernetics, Part B. Cybernetics: A publication of the IEEE Systems, Man, and Cybernetics Society,2012,42(1):218-233.
[11] Gao X, Ji Q, Mi Z, et al. Similarity Measure based on Punishing Popular Items for Collaborative Filtering[C]// 2018 International Conference on Computer, Information and Telecommunication Systems (CITS). IEEE, 2018.
[12] Hao Li-yan, WANG Jing. Collaborative Filtering TopN recommendation Algorithm based on Item Popularity [J] .Computer Engineering and Design, 2013, 34(10): 3497-3501.
[13] Wei Tian-tian, Chen Li, Fan Ting-ting, Wu Xiao-hua. Collaborative Filtering recommendation Algorithm based on Item Popularity Weighting [J]. Application Research of Computers, 2020, 37(03): 676-679.
[14] Yi D, Xue L. Time weight collaborative filtering. ACM, 2005.
[15] Chen Y C, Hui L, Thaipisutikul T, et al. A Collaborative Filtering Recommendation System with Dynamic Time Decay[J]. The Journal of Supercomputing, 2020:1-19.
[16] Song Wei-wei, Yang De-gang, Zheng Min. Research on collaborative Filtering recommendation Algorithm based on time weighted label[J]. Journal of Chongqing Normal University (Natural Science), 2016, 033(005):113-120.