Multiple similarity collaborative filtering recommendation among users

Wenjing Yan*, Shuqing Li and Yongshang Cheng

School of Information engineering, Nanjing University of Finance and Economics, Nanjing, China

*Corresponding author e-mail: 844569289@qq.com

Abstract. [Objective] Through the analysis of multiple similarity among users, the problem that the traditional user based collaborative filtering algorithm only uses a single similarity and leads to the decline of recommendation quality is solved. [Method] The original single similarity calculation formula is improved, and the multiple similarity calculation formula is put forward, on this basis, the multiple similarity prediction score is calculated. [Result] By comparison with the traditional user based collaborative filtering algorithm, the method put forward in this paper has outstanding effect. [Limited] Users' interests will change with time, so time information should be included in the calculation. [Conclusion] From the experiment, we can find that the improved method has better recommendation quality than traditional methods.

1. Introduction

In such a rapidly changing era, the way people access information has greatly changed. However, the increasing amount of Internet information makes the problem of information overload more and more serious [1]. As a technical means to effectively solve the information-overloading problem, the recommendation system is receiving more and more attention from the industry. Collaborative filtering is a very mature and popular recommendation algorithm in practice [2]. Most of the current collaborative filtering recommendation algorithms mainly recommend to a user by calculating a user's forecast score of the non-scored items and taking this as the main basis. Among them, the traditional algorithm bases the calculation of the predicted score on the real score records of other users similar to the current user. The so-called "similar" users tend to give similar scores when rating the project, and this is also the main basis for the use of collaborative filtering ideas for scoring prediction.

However, the traditional algorithm algorithm itself, as well as the known improvements based on the algorithm, all adopt a single similarity to describe the similarity between users for any project.
preference, without considering that different types of projects may have different similarity. However, daily experience tells us that people tend to have different preferences for different types of things. That is to say, two people who are interested in one thing at the same time may have different degrees of preference for another type of thing. In the scoring of the project, the scores of two people for a certain type of project are relatively close, but the scores for another type of project may be very different. Therefore, from this point of view, the traditional user based collaborative filtering algorithm only depends on a single similarity to measure the similarity of two users' preferences for all types of projects is not very reasonable.

2. Related research work

2.1. Collaborative Filtering Recommendation Algorithm
The collaborative filtering recommendation algorithm excavates the user's preferences by mining the past data of the user, and sorts the users according to various tastes and recommends similar products. The collaborative filtering recommendation algorithm can be divided into a user-based collaborative filtering recommendation algorithm and an item-based collaborative filtering recommendation algorithm.

(1) User-based collaborative filtering algorithm
User-based collaborative filtering algorithm, the core principle is to use users who have similar interests with the target user u and have scored the same item. These users are called neighbors of user u. In general, the more items the two users have scored together, the more likely the user is to be selected as the nearest neighbor of the target user [3].

(2) Item-based collaborative filtering algorithm
Item-based collaborative filtering algorithm, the similarity between items is calculated and recommend to the user something similar to his favorite item. This method is based on the item's score to calculate the similarity, rather than analyzing the keyword vector similarity [4].

The main idea of the collaborative filtering algorithm is to construct a collaborative filtering matrix based on the user's existing past data, where each row of data represents the user's rating on all items; Each column represents the rating obtained by the project. Since the algorithm defaults to users with similar browsing history and scoring behavior, it is often similar in subsequent project selection and evaluation.

2.2. User similarity evaluation index
The similarity calculation in the collaborative filtering recommendation algorithm is a very important step. The following are several common methods used to calculate the similarity between users or items [5].

(1) Pearson Correlation Coefficient

\[
Pearson(u,v) = \frac{\sum_{i \in I} (r_{ui} - \bar{r}_u)(r_{vi} - \bar{r}_v)}{\sqrt{\sum_{i \in I} (r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{i \in I} (r_{vi} - \bar{r}_v)^2}}
\]  

(1)

Which \( r_{ui} \) represents the score of user u on item i, and \( \bar{r}_u \) represents the average of the user's ratings.
The coefficient is between -1 and 1, where -1 is the least similar and 1 is the most similar.

(2) Cosine Similarity

\[
\cos(u, v) = \frac{\sum_{r \in U} r_{ui} \cdot r_{vj}}{\sqrt{\sum_{r \in U} (r_{ui})^2} \sqrt{\sum_{r \in U} (r_{vj})^2}}
\]

Among them, \(r_{ui}\) represents the score of user \(u\) on item \(i\), and the Cosine similarity value ranges from 0 to 1.

(3) Jaccard Similarity

\[
\text{Jaccard}(u, v) = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|}
\]

Which \(N(u)\) represents the number of user ratings, Jaccard similarity value ranges from 0 to 1.

(4) Euclidean Distance

\[
d(u, v) = \frac{1}{1 + \sqrt{\sum_{r \in I} (r_{ui} - r_{vj})^2}}
\]

Which \(r_{ui}\) represents the score of user \(u\) on item \(i\), and the value of Euclidean distance ranges from 0 to 1.

2.3. Multiple similarity collaborative filtering recommendation among users

The traditional algorithm only depends on a single rating similarity to compute the user's prediction score for unrated items, which is not reasonable. This single scoring similarity is based on the similarity of all the historical scoring records of two users, but it can not reflect the similarity of their real preferences for a certain type of project, so it reduces the recommendation quality of the algorithm to a certain extent [6, 7].

In real life, people's preferences for different types of things are often different. It is based on this observation that multiple similarities between users are proposed [8, 9]. The multiple similarity between users refers to the multiple independent scoring similarity between users on different project types. The similarity between users about a certain project type is calculated based on the scoring records of the two users for the projects of this type, Therefore, this similarity can more exactly reflect the similarity between the two users' preferences for this type of project. In this way, by calculating the similarity scores of users independently for different types of items, then we can use multiple similarity to more accurately depict the similarities and differences of users' interests and tastes for various types of projects, and get more accurate prediction score on this basis.

Here, let the set \(T = \{t_1, t_2, ..., t_k\}\) represent all types of sets, and each item can belong to one or more types in the set \(T\). In order to describe the different preferences of users for these \(k\) types of items, the similarity between users based on all \(k\) types of items needs to be calculated. For any two users \(X\)
and Y, their k similarities for each type in the set T are expressed as \( \text{Sim} (x, y, t_1), \text{Sim} (x, y, t_2), \ldots, \text{Sim} (x, y, t_k) \). Then,

\[
\text{Sim}(x, y, t_m) = \frac{\sum_{i \in I_{x,y}^m} (r_{x,i} - \bar{r}_x^m) (r_{y,i} - \bar{r}_y^m)}{\sqrt{\sum_{i \in I_{x,y}^m} (r_{x,i} - \bar{r}_x^m)^2} \sqrt{\sum_{i \in I_{x,y}^m} (r_{y,i} - \bar{r}_y^m)^2}} \quad (5)
\]

When calculating the prediction score based on multiple similarities among users, the K-neck method is also used. Different from the traditional algorithm, because it takes into account that users have different similarities for different types of items, and usually an item can belong to multiple types, Therefore, when calculating the current user's prediction score for a certain project, first of all, we need to select k users with the highest similarity with the current user about each type of the project as the nearest, to compute the user's multiple prediction scores based on each type of the project, and finally take their weighted average as the final prediction score [10].

2.4. Algorithm flow

This paper proposes a collaborative filtering recommendation algorithm based on multiple similarities between users, that is, by calculating the similarity of ratings between users for different project types, a more accurate prediction score for users for different types of projects is obtained. The specific flow of the algorithm is shown in Table 1:

| Algorithm: Multiple similarity collaborative filtering algorithm |
|---------------------------------------------------------------|
| **Input:** user rating matrix R                                |
| **Output:** prediction score                                   |
| **Process:**                                                   |
| (1) Calculate the multiple similarity matrix between users     |
| (2) User's prediction matrix for movie                         |
| (3) Weighted average to get the final prediction score         |

3. Experiment

3.1. Data set

This paper uses the Movielens 100K dataset to test the difference in recommendation quality between the improved multiple similarities algorithm and the traditional recommendation algorithm. The data set recorded 100,000 ratings of 1,682 movies by 943 users. Every user rated more than 20 movies, and the scores are between 1 and 5.

3.2. Evaluation Criteria

Mean absolute error is a reflection of the difference between two variables. It is a common indicator for measuring the accuracy of recommendation algorithms. It is also a very common evaluation indicator in the field of recommendation systems. The calculation formula is shown in (6)
MAE = \frac{\sum_{u,t \in T}|\hat{r}_{ut} - r_{ut}|}{|T|} \tag{6}

3.3. Experimental results

The experiment randomly selected the score records of 100, 500, and 1000 users from the data set as the test data set, and further randomly selected 80% of the score records as the training set and the other 20% as the test set. During the experiment, MAE was used as the criterion for evaluating the recommendation algorithm. The number of neighbors was taken from 30 to 50 and the interval was 5. The following figures are the experimental results:

**Figure 1.** Comparison of MAE under 100 users.

**Figure 2.** Comparison of MAE under 500 users.
It can be seen from the figures that when using a dataset of 100 users for experiments, in the case of using related similarity to compute the similarity between users, as the number of neighbors increases, whether it is a User-Based collaborative filtering recommendation algorithm or a multiple similarity collaborative filtering algorithm, their MAE values show a downward trend, but no matter how many neighbors are selected, in the situation of the same number of neighbors, the collaborative filtering recommendation algorithm based on multiple similarity between users has smaller MAE value. Similarly, experiments on datasets of 500 users and 1000 users also yielded similar results. This shows that in the case of calculating user similarity based on Pearson similarity, compared with the traditional algorithm, the algorithm based on multiple similarities among users can achieve better recommendation results.

4. Conclusions
In consideration of the drawback in the traditional User-Based collaborative filtering recommendation algorithm, this paper proposes a collaborative filtering recommendation algorithm based on multiple similarities between users, that is, by calculating the similarity between users for different project types, we can get more accurate prediction scores for different types of projects. From the experimental results, we can find that compared with the user-based collaborative filtering recommendation algorithm, this algorithm can more accurately recommend items of interest to target users and improve the efficiency of the algorithm's recommendation. However, there are still some areas for improvement in this algorithm, mainly including the following two points:

(1) The interest of users will change with time, and time information should be included in the calculation.

(2) Failed to solve the problems of cold start and data sparseness in collaborative filtering.
References

[1] XIONG H, LIU Z. A situation information integrated personalized travel package recommendation approach based on TD-LDA model[C]//Behavioral, Economic and Socio-cultural Computing (BESC), International Conference on. IEEE, Nanjing, China, 2015:32-37.0.

[2] Xin M, Zhang Y, Li S, et al. A Location-Context Awareness Mobile Services Collaborative Recommendation Algorithm Based on User Behavior Prediction[J]. International Journal of Web Services Research, 2017, 14(2):45-66.

[3] Wang Guang, Jiang Li, Dong Shuaihan, Li Feng. Collaborative filtering algorithm combining ontology semantics and user attributes [J]. Computer Engineering, 2019, 45 (10): 215-220.

[4] Wang Dong, Xue Feng, Liu Kai, Chen Siyang, Zhang Haobo. A Collaborative Filtering Recommendation Algorithm Based on Item Fusion Autoencoder [J]. Computer Application.

[5] Wenxia Lu, Lijun Zhang. Collaborative block-delay Volterra filters for nonlinear acoustic echo cancellation[J]. Applied Acoustics, 2019, 156.

[6] Engineering; New Engineering Study Results Reported from Xiamen University (Deep Probabilistic Matrix Factorization Framework for Online Collaborative Filtering)[J]. Computers, Networks &amp; Communications, 2019.

[7] Networks - Neural Networks; Findings from Northeast Normal University in the Area of Neural Networks Reported (Trust-aware Collaborative Filtering With a Denoising Autoencoder)[J]. Computers, Networks &amp; Communications, 2019.

[8] Song Yongjian, Song Jinling, Zhang Zhengyang, Xu Jiasong, Zhao Jialin. A collaborative filtering recommendation algorithm based on item score prediction and user similarity [J]. Modern Agriculture, 2019 (03): 60-62.

[9] Lu Hang, Shi Zhibin, Liu Zhongbao. Collaborative Filtering Recommendation Algorithm Fusion of User Interests and Rating Differences [J / OL]. Computer Engineering and Applications: 1-8 [2019-11-26]. http://kns.cnki.net/kcms/detail/11.2127.tp.20191115.1136.004.html.

[10] Cao Yin, Li Wei. Collaborative filtering recommendation algorithm based on trust and graph clustering [J]. Information Technology, 2019 (10): 77-82.