Journal of Physics:
Conference Series

PURPOSE-LED
PUBLISHING™

PAPER • OPEN ACCESS

Improved collaborative filtering recommendation algorithm based on user attributes and K-means clustering algorithm

To cite this article: Lihong Chen *et al* 2021 *J. Phys.: Conf. Ser.* **1903** 012036

Improved collaborative filtering recommendation algorithm based on user attributes and K-means clustering algorithm

Lihong Chen[1], Yi Luo[2], Xudong Liu[3], Weijie Wang[4], and Ming Ni[1*]

[1]College of Information Engineering, Sichuan Agricultural University, Ya’ an, Sichuan, Province, 625000

[2]College of Science, Sichuan Agricultural University, Ya’ an, Sichuan, Sichuan,625000

[*]Corresponding author’s e-mail: nm@sicau.edu.cn

Abstract. Aiming at the problem of poor performance of collaborative filtering algorithm on data sets with large sparsity, this paper proposes an improved collaborative filtering recommendation algorithm which integrates user attributes and K-means clustering. When considering user similarity, the weight of user attributes is introduced to reduce the impact of data sparsity on similarity calculation. Meanwhile, the characteristics of user's age, gender and occupation are concerned. At the same time, combined with K-means clustering, the algorithm can further improve the accuracy of the recommendation model.

1. Introduction

Collaborative filtering recommendation algorithm is one of the earliest and most widely used recommendation algorithms. The core of the algorithm is to find users' preferences by mining users' historical behavior data, divide users into groups based on different preferences, and recommend other users' preferences to users in the same group.

In reality, the number of product sets M is much larger than the number of user sets N, so the sparsity of user rating matrix $R_{ui}^{N\times M}$ generated from user historical behavior data is very high, and the accuracy of user group division is low. At this time, the effect of collaborative filtering algorithm recommendation model is poor[1]. In order to improve the accuracy of finding the nearest neighbor users, some literatures use k-means clustering algorithm[2]. The algorithm firstly classifies the user data according to the characteristics of the user data itself, and then uses collaborative filtering algorithm for recommendation. This method improves the recommendation effect.

However, for the new users or inactive users with less data, it is still unable to improve the difficulty of dividing user groups and the low accuracy of finding the nearest neighbor set. Therefore, this paper proposes an improved collaborative filtering recommendation algorithm which integrates user attributes and K-means clustering. The core change of the algorithm is the introduction of user attributes {age, gender, Occupation} when calculates the distance between users. Give the user attributes weight {a_1, a_2, a_3} in the calculation formula. Calculate the user similarity twice, which reduces the impact of sparse user history score data on finding nearest neighbor users, increases the impact of user characteristics on group division, enhances the accuracy of finding nearest neighbor set, and improves the efficiency of recommendation model.

2. Traditional Collaborative Filtering Methods

2.1. User-based Collaborative Filtering Recommendation Algorithm

In the history of recommend system, collaborative filtering recommendation algorithm based on user is the earliest. Based on the idea of "birds of a feather flock together, people flock together", the algorithm recommends the favorite items of neighbor users with high similarity to the target users. User- based collaborative filtering recommendation algorithm needs to establish user item scoring matrix[3], set user = {u_1, u_2... u_N}, item = {i_1, i_2... i_m}

Table 1 user project scoring matrix

	i_1	i_2	i_3	...	i_m
u_1	r_{11}	r_{12}	r_{13}	...	r_{1m}
u_2	r_{21}	r_{22}	r_{23}	...	r_{2m}
u_3	r_{31}	r_{32}	r_{33}	...	r_{3m}
...	...	...	...	...	...
u_v	r_{v1}	r_{v2}	r_{v3}	...	r_{vm}
...	...	...	...	...	...
u_n	r_{n1}	r_{n2}	r_{n3}	...	r_{nm}

The algorithm consists of two steps:

- The user-item rating matrix is established, and the user similarity sim (u, v) is calculated according to the scoring vector {r_{v1}, r_{v2}, r_{v3}... r_{vm}} of each user, and the user set similar to the interest of the target user is found, that is, the nearest neighbor user set N_u.
- Find the scoring items of the users in the nearest neighbor set, and give weight to the items that have not been scored by the target users.

$$w_i = \sum_{v \in N_u} r_{vi} \times sim(u, v) \tag{1}$$

The first n items are recommended according to the weight of the project.

2.2. Item-based Collaborative Filtering Recommendation Algorithm

Item-based collaborative filtering recommendation algorithm[4] is one of the most basic and simple recommendation algorithms, and its steps are similar to those based on users.

But the difference is that the CF recommendation algorithm based on item calculates the similarity between items by establishing the item user behavior operation matrix, such as item user rating matrix, item user collection matrix, item user browsing time matrix, etc., predicts the score according to the item similarity, and pushes the items that are similar to the items that the target user likes Recommend to target users, this recommendation algorithm often recommends the same type of items, the recommendation accuracy is high, but the lack of surprise.

3. Hybrid Recommendation Algorithm based on K-means Clustering and User Attributes

3.1. Collaborative Filtering Recommendation Algorithm based on K-means

3.1.1. K-means clustering algorithm

K-means is an unsupervised machine learning method, which can be divided into K clusters according to the characteristics of the data itself, so that the distance between the points in the cluster is smaller, and the distance between the points in the cluster is larger, so the similarity between the points in the cluster is larger, but the similarity between the points in the cluster is smaller.

Algorithm flow:

- Randomly initialize K centroids {c_1, c_2,... c_K}.

- The Euclidean distance formula calculates the distance from each particle to the center of mass:

$$dis(c_k,u_j)=\sqrt{\sum_{i=1}^{m}(r_{ki}-r_{ji})^2} \tag{2}$$

and the particles are divided into clusters with the smallest distance centroid.

- Calculate the mean vector of K clusters as the next round centroid {c_1 ', c_2',... c_k '}.
- Repeat step 2 until the sum of the distances from each particle to the center of mass is constant or less than a minimum.
- K cluster classifications are obtained.

In the case of large data sparsity, the k-means algorithm is used to divide users into K clusters before collaborative filtering, which reduces the difficulty of finding the nearest neighbor set and improves the accuracy of the search[5].

3.1.2. Improved Collaborative Filtering Algorithm based on K-means

The traditional collaborative filtering recommendation algorithm only uses the calculation formula to calculate the user similarity, and the efficiency is very low when the user data scale is large. After clustering the users with k-means algorithm, it not only improves the accuracy of recommendation, but also reduce the nearest neighbor search space of the target user.and improve the scalability of the algorithm by determining the cluster of the target user, calculate the similarity between the users in the cluster and the target user.

The user based collaborative filtering algorithm needs to calculate the similarity between user u and all other users {v_1,v_2...v_{N-1}}. When the user score is less, the effect is poor. After the first K-means clustering, it only needs to calculate the similarity between user u and the user {v_1, v_2...v_q} in the cluster {k_u:(v_1,v_2...u...v_q)},which improves the computational efficiency and accuracy.

Algorithm flow:

- K-means clustering algorithm is used to get K clusters.
- Cosine distance is used to calculate the similarity between user u and user {v_1, v_2...v_q} in the cluster

$$sim(u,v)=\frac{\sum_{i=1}^{M} r_{ui}r_{vi}}{\sqrt{\sum_{i=1}^{\mathrm{M}} r_{ui}}\times\sqrt{\sum_{i=1}^{\mathrm{M}} v_{vi}}} \tag{3}$$

M is the number of items, using the top n users with the highest similarity {v_1,v_2...v_n} as the nearest neighbor users.

- Add the items that the user has not scored to the recommendation list, and give weight to them.

$$w_i=\sum_{j=1}^{\mathrm{n}} sim(u,v_{\mathrm{j}})\times r_{v_j i} \tag{4}$$

Where is the similarity between the target user u and the user in the nearest neighbor set, and is the score of the user in the nearest neighbor set on item

- Get the recommendation list of user u.

3.2. Collaborative Filtering Recommendation Algorithm based on User Attributes

When the user u score is less or no score, the calculated similarity between users is very small, it is often difficult to find the nearest neighbor users, and the accuracy of the recommended items is also low, so the collaborative filtering algorithm has poor effect when the user data is sparse. In view of this situation, this paper uses the collaborative filtering algorithm combined with user attributes to solve the problem. When calculating the similarity between users, we introduce the user attribute, age, gender, occupation and give them the weights {a_1, a_2, a_3}. The similarity calculation formula [6] is as follows:

$$\mathrm{simim}(u,v)=(1-a_1-a_2-a_3)\times sim(u,v)+a_1\times\frac{1}{|\,age(u)-age(v)\,|+1} \tag{5}$$
$$+a_2\times(sex(u)\bigcap sex(v))+a_3\times(major(u)\bigcap major(v))$$

$age(u)$ is the age of user u, $age(v)$ is the age of user v; $sex(u)$ is the gender of user u, $sex(v)$ is the gender of user V, when the gender is the same, $sex(u)\bigcap sex(v)=1$; $major(u)$ is the specialty of user u, $major(v)$ is the specialty of user V, when the specialty of user is the same, $major(u)\bigcap major(v)=1$; when the calculation $\mathrm{sim}(u,v)$ is very small due to the sparse user rating data, by adjusting the proportion of user attributes $\{a_1,a_2,a_3\}$ in order to reduce the impact of data sparsity on finding the nearest neighbor users and improve the accuracy of recommendation.

3.3. Improved Collaborative Filtering Recommendation Algorithm integrating User Attributes and K-means Clustering

In order to further improve the accuracy of finding the nearest neighbor users and recommending items, this paper uses K-means clustering algorithm and user attribute combination method_ The algorithm flow is as follows:

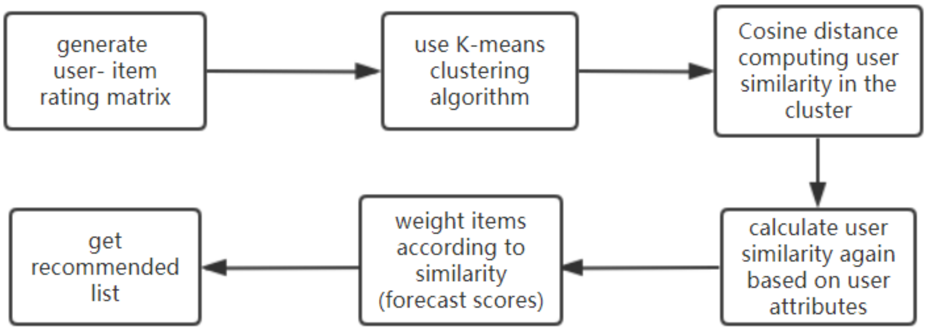
generate user- item rating matrix
use K-means clustering algorithm
Cosine distance computing user similarity in the cluster
calculate user similarity again based on user attributes
weight items according to similarity (forecast scores)
get recommended list

Figure 1 our algorithm flow chart

Firstly, the user-item rating matrix is established. According to the user score vector, K-means clustering algorithm is used to classify, and K clusters are obtained. Then cosine similarity is calculated in the cluster to which the target user belongs. Secondly, user similarity is calculated by combining with user attributes {age, gender, occupation}, and the nearest neighbor user set is obtained. Finally, project evaluation is predicted according to the user similarity in the nearest neighbor set Points, get the recommended list.

4. Experimental Results and Analysis

4.1. Experimental Data Set

The experiment uses the Movielens data set released by the University of Minnesota, including the rating data of multiple users (1-5), movie metadata information and user attribute information. The details are as follows:

Table2. Details about the data set.

number of users	number of movies	number of ratings	user attribute1	user attribute12	user attribute13
943	1682	100000	Age	Gender	Occupation

These data are collected through Movielens website. Each user has scored at least 20 movies, including movie information (movie title, release date, video release date), movie classification information and user attributes (age, name, occupation, zip code). This paper only uses age, gender, occupation in user attributes, and the data set we used is 100k.

4.2. Evaluation Index

In this paper, 80% of the scoring data in the data set is used as the training set, 20% as the test set, and the mean absolute error MAE is used as the evaluation index of the recommended model:

$$MAE = \frac{1}{\mathrm{n}}\sum_{i=1}^{n}\left|p_{ui} - real_{ui}\right| \tag{6}$$

Among them, p_{ui} is the recommendation model to predict the user's score of the item, $real_{ui}$ is the user's real score of the item in the test set[8]:

$$p_{ui} = \overline{r_u} + \frac{\sum_{j\in I} simim(u, v_j) \times \left|r_{vi} - \overline{r_{v_j}}\right|}{\sum_{j\in I} simim(u, v_j)} \tag{7}$$

$\overline{r_u}$ is the average score of user u, $\overline{r_{v_j}}$ is the average score of neighbor users, I is the user set that has scored item I in neighbor set n, and the neighbor user set that has scored item I in neighbor set n.

4.3. Experimental Results

4.3.1. The influence of cluster number on Evaluation Index:

In the calculation of user similarity, the weight setting of user attribute is {a1=0.2, a2=0.2, a3=0.2} that is, {age = 0.2, gender = 0.2, Occupation = 0.2}, that is, the influence weight of user attributes on the similarity between users, the proportion of user similarity weight calculated by cosine distance is 1-a1-a2-a3 = 0.4; the number of adjacent users is n = 10, K is the number of clusters, which determines the number of central particles and data in K-means clustering algorithm is divided into several categories, K is the independent variable, and the absolute error MAE is the dependent variable:

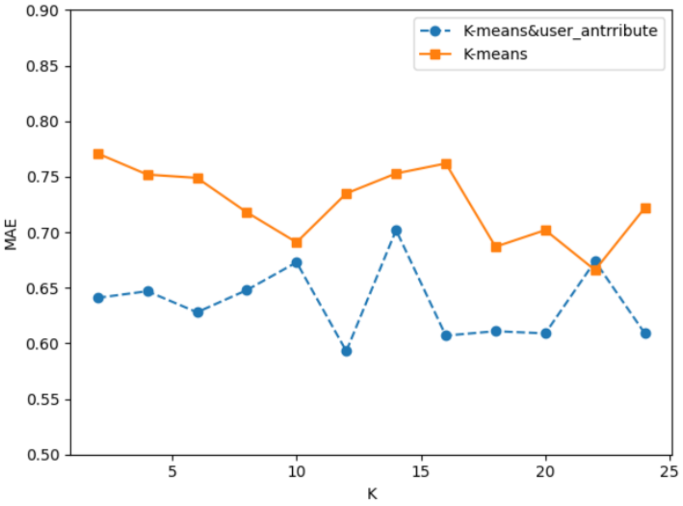

Figure2. Comparison of K-means clustering CF algorithm and K-means clustering CF algorithm with user attributes

4.3.2. The influence of user attribute weight on the evaluation index:

When k = 12, the absolute error is the smallest, so the number of clusters = 12 is used for the experiment. The number of nearest neighbor users is 10, and the value of user attribute weight {A1, A2, A3} is selected as the independent variable, that is, the influence proportion of user attribute {age, gender, occupation} on user similarity calculation:

Table3. Effect of user attribute weight on MAE.

a1	a2	a3	MAE
0.05	0.05	0.05	0.687
0.1	0.1	0.1	0.639
0.15	0.15	0.15	0.710
0.2	0.2	0.2	0.644
0.25	0.25	0.25	0.675
0.3	0.3	0.3	0.582

4.3.3. Experimental Analysis

It can be seen from Figure 2 that the overall recommendation accuracy of the algorithm used in this paper is based on the K-means clustering CF algorithm and the simple CF algorithm. The difference is that the algorithm used in this paper introduces the user attribute weight, calculates the similarity between users twice, and further improves the accuracy of finding the nearest neighbor users. The sparse degree of Movielens data set is large, which leads to small user similarity. However, the introduction of user attribute weight can reduce the impact of higher sparsity on user similarity. As can be seen from Table 3, when the proportion of user attribute in similarity calculation weight $a_1+a_2+a_3$ is the largest, the recommendation effect is better.

5. Conclusion

Aiming at the poor recommendation effect of collaborative filtering algorithm in the case of high sparsity of user rating matrix, this paper proposes an improved collaborative filtering recommendation algorithm combining user attributes and K-means clustering, which introduces user attribute weight to improve the accuracy of recommendation model. However, for the common cold start problem in recommendation scenario, it is still difficult to find the nearest neighbor set when there are few kinds of user attributes, and the effect of recommendation model is poor.

Acknowledgment

Funding: This research was supported by the Research Interest Training Program of Sichuan Agricultural University (Grant No.2020636) and the Sichuan Agricultural University (Grant No. X2019076).

References

[1] Wang Yong, Wang XiaoY, Tao Yazhi, Zhang Pu (2017) Collaborative filtering recommendation algorithm based on K-medoids item clustering[J]. Journal of Chongqing University of Posts and Telecommunications(Natural Science Edition),2017,29(04):521-526.

[2] Lin Ruyi, He Feng, Zhao Xiaolong. (2017) Collaborative filtering algorithm based on K-means clustering[J]. Computer Fujian,2017,33(12):6-8.

[3] Wang Sanhu, Wang Fengjin. (2017) A collaborative Filtering recommendation algorithm based on user score and antribute similarity[J]. Computer Application and Software, 2017, 34(04): 305-308+321.

[4] Chigozirim Ajaegbu. (2021) An optimized item-based collaborative filtering algorithm[J]. Journal of Ambient Intelligence and Humanized Computing,2021(prepublish).

[5] Zhao Wei, Lin Nan, Han Ying, Zhang Hongtao. (2016) User-based collaborative filtering recommendation algorithm based on improved K-means clustering[J].Journal of Anhui University(Natural Sciences), 2016, 40(02): 32-36.

[6] Xia Jingming, Liu Conghui. (2020) A collaborative filtering algorithm based on user and commodity attribute mining[J]. Modern Electronics Technique,2020,43(23):120-123.

[7] Li Yanjuan, Niu Mengting, Li Linhui. (2019) A collaborative filtering recommendation algorithm based on a bee colony K-means clustering model[J]. Computer Engineering and Science, 2019, 41(06):1101-1109.