Collaborative Filtering Recommendation Algorithm for User Interest and Relationship Based on Score Matrix

Kejia Xue and Junyi Wang[*]
Inner Mongolia University of China, China
[*]Corresponding author

***Abstract*—An improved collaborative filtering recommendation algorithm is proposed to solve the problem of sparse and low recommendation accuracy of traditional collaborative filtering recommendation algorithm. User preferences and user trust relationships are used to calculate the user's preferences for the project, and the user ratings are used to fill the scoring matrix with unrated items. Considering the change of user interest and user relationship, we introduce time based interest weight function and preference degree to the project similarity computation and recommendation process, and identify the nearest neighbor set, so as to achieve the best recommendation. User preferences and user trust relationships are used to calculate the user's preferences for the project, and the user ratings are used to fill the scoring matrix with unrated items.**

Keywords—collaborative filtering; trust; user interest; sparse data; matrix filling

I. Introduction

The rapid development of information technology has brought great convenience to people, but also brought about information overload. In order to solve this problem, many researchers have put forward many effective solutions. Recommendation system as one of the representative type of solution link to different users and different items by the personalized recommendation algorithm. It is convenient for users to easily find themselves interested in things, and can efficiently show items to interested users. Personalized recommendation technology through information filtering to solve the above problems, the application of a wide range of collaborative filtering algorithm -- K neighboring algorithm[1] mainly includes the following steps:

a) similarity matrix is obtained using the similarity calculation formula.

b) to obtain *the user (project) K near neighbor.*

c) the scoring project is predicted by the scoring prediction formula, and the corresponding recommendation set is generated.

However, there are still some shortcomings in the k-nearest neighbor algorithm, such as the low recommendation accuracy due to data sparsity, the cold start-up of users (project) and the recommendation of diversity[2].

In view of the above problems, scholars have carried out a series of improvement studies. Wang Jun, de Vries A P and Reinders M J T propose a unified collaborative filtering algorithm based on user and project-based similarity[3], which can reduce the data sparsity and improve the accuracy of the nearest neighbor search. Huang Chuangguang and his colleagues propose a collaborative filtering recommendation algorithm is proposed, which can be used to select the prediction target, and the trust subgroup with high trust is recommended to improve the prediction accuracy.Xu Zhihong and Wang Baoying propose the formula of emissivity of heat energy is applied to the calculation formula of comprehensive similarity to improve the accuracy of recommendation.

In the above literature, the similarity calculation method is obtained through the common score item set between users. However, in the case of extremely sparse data, the user's common scoring project set may be small, which results in a low accuracy of the similarity calculation. The user similarity measure is closely related to the user's interest, in addition to the user's rating of the project. Xing Chunxiao and his colleagues calculate the weight based on time window and the similar data weight, and uses linear attenuation function to determine the user's interest over time.By adding the user's trust and attribute information of the project, Chen Zhiming and Li Zhiqiang use the time strategy of interest change based on the rule of forgetting to recommend the user to the neighbor collection. On this basis, we propose a collaborative filtering recommendation algorithm based on scoring matrix filling and user interest. Using the preference relation with users to populate the sum of the weighted score users never score projects to solve the data sparseness through comprehensive user preference for the project properties and the relationship between user computing user preference for a project. At the same time, the time weight function is added to the similarity calculation and recommendation, and the recommendation accuracy is further improved considering the changes of users' interests.

II. Traditional Collaborative Filtering Algorithm

Set user assemble U = $\{U_1, U_2,\ldots,U_m\}$, project assemble I = $\{I_1, I_2,\ldots,I_n\}$, $R_{m\times n}$ (M is the number of users, n is the number of items) to represent the user's rating matrix for the project. r_{ui} is the score of user U for project I (the score is between 1 and 5) and the "0" value in the matrix indicates is that a user did not grade the project. The main steps of the user-based collaborative filtering algorithm are as follows:

Step 1: Solve the similarity matrix between project and project.

Collaborative filtering algorithm first finds the current user similarity items based on the score matrix. It is commonly used to calculate the connection degree between 2 fixed distance variables According to the above Pearson correlation coefficient. This similarity method is used to calculate the similarity of user i and j:

$$\mathrm{sim}(i,j)=\frac{\sum_{u\in R(i)\cap R(j)}(r_{ui}-\overline{r_i})\times(r_{uj}-\overline{r_j})}{\sqrt{\sum_{u\in R(i)\cap R(j)}(r_{ui}-\overline{r_i})^2}\times\sqrt{\sum_{u\in R(i)\cap R(j)}(r_{uj}-\overline{r_j})^2}} \quad (1)$$

$\mathrm{u}\in R(i)\cap R(j)$ represents users who have been graded for both project i and j; r_{uj} is the rating of user u for project J; r_i represents the average score of users on project I; r_j represents the user's average score for project j.

Step 2: Solving project K nearest neighbor set

The similarity matrix of the target project is calculated by equation (1), and then descending order, and the former K maximum value is denoted as KNN(i) :

$$KNN(i)=\{I_j\,/\,sim(i,j)\in S(i)_{k\max}\} \quad (2)$$

In which, I_j represents a set of projects; $S(i)_{k\max}$ represents the top K value of the highest similarity to the project I.

Step 3: Generate recommendation sets

Set the nearest neighbor set of the target user a to be KNN (i), then the user a's prediction score for the non rated project $\hat{r}_{ui}$ can be obtained by formula (3).

$$\hat{\mathrm{r}}_{ui}=\frac{\sum_{j\in R_i^K(u)}\mathrm{sim}(i,j)\times r_{uj}}{\sum_{j\in R_i^K(u)}sim(i,j)} \quad (3)$$

Among them, $\mathrm{j}\in R_i^K(u)$ indicates that j is the closest neighbor of K to i.

III. Collaborative Filtering Recommendation Algorithm Based on Score Matrix Filling and User Interest

A. *User Preference*

The traditional collaborative filtering algorithm generally considers the user dominant score as the user's preference, but because the scoring matrix is too sparse, it can't get the nearest neighbor accurately. Considering that the interest of users in real life is mainly embodied in one or some category, that is, the preference of category attributes for a project or a commodity. At the same time, the relationship between users will also affect user preferences. Therefore, the user's preference for the project is calculated from the perspective of user and user relations.

1) User attribute preference: With the expansion of the scale of e-commerce, the number of users and projects of the system has increased rapidly, but the existing classification information of the project has not changed. Each project has its own attributes that allow users to quickly search for what they need[4]. In general, the item - attribute information of the commodity in the e-commerce system is shown in Figure I.

	A_1	A_2	A_3	…	A_p
I_1	S_{11}	S_{12}	S_{13}	…	S_{1n}
I_2	S_{21}	S_{22}	S_{23}	…	S_{2n}
I_3	S_{31}	S_{32}	S_{33}	…	S_{3n}
…	…	…	…	…	…
I_n	S_{n1}	S_{n2}	S_{n3}	…	S_{nn}

FIGURE I. USER-PROJECT MATRIX

In Figure I, I represents the project number; A is a project attribute; S_{ij} indicates whether the item has an attribute of Aj, with a value of 1 or 0. If the item I has an attribute Aj, then its value is 1, otherwise 0. For a user, the user's preference for different attribute features can be measured according to the attribute matrix mentioned above:

$$p(u,j)=\frac{size(S_{u,j})}{size(I_u)} \quad (4)$$

Among them, $size(S_{u,j})$ indicates that the user u is evaluating the total number of J class projects; $size(I_\mathrm{u})$ represents the number of all item sets that the user u has scored.

2) User trust calculation: Using the traditional collaborative filtering algorithms recommend items for the user, the user can often be affected by the score matrix of data sparsity, unable to calculate the similarity of users, using compute trust between users instead of user similarity, can to some extent alleviate the problem of data sparseness.

In this paper, trust degree will be used as a measure of a user's prediction accuracy. The calculation of trust value is carried out among every two users, and the trust relationship is irreversible. That is, the user A is highly trusted with the user B, and the user B is still highly likely to be untrusted to the user A. For a target user, the user may also receive from many other users recommend items, but eventually choose to buy items and scoring is just part of them, when the target users of the goods can be the actual score, score and rating prediction target users are compared, and the trust between them value. In the whole process, we only focus on the prediction accuracy of the recommended user v corresponding to the target user u, which is actually purchased and evaluated by v. In order to evaluate the prediction accuracy of the recommended user v, we first use the formula (5) to generate the score prediction $\hat{r}_{u,i}$ of the recommended user v.

$$\hat{r}_{u,i} = r_{v,i} + (\bar{r}_u - \bar{r}_v) \tag{5}$$

Among them: $r_{v,i}$ represents the actual score of the recommended user v for the item i.After the prediction score of the recommended user v, the prediction accuracy of the recommended user v relative to the target user u on the item I $p_v(u,i)$ is calculated by the formula (6):

$$p_v(u,i) = 1 - \frac{\left|\hat{r}_{u,i} - r_{u,i}\right|}{4} \tag{6}$$

Where $p_v(u,i)$ is the actual score of the target user u on item i, the recommendation system to score 5 points, the 5 represents the highest score of 5 points, 1 represents the lowest score of 1 points, the highest score and the lowest score here using the difference 4 as the denominator to ensure the value $p_v(u,i)$ remains in the range of [0,1] , and the greater $p_v(u,i)$,the higher prediction accuracy indicates the score.Then the prediction accuracy of the score is magnified to the global, and the user u's trust degree to the user v $trust(u,v)$ is generated by the formula (7):

$$trust(u,v) = \frac{\sum_{i \in I_{u,v}} p_v(u,i)}{\left|I_{u,v}\right|} \tag{7}$$

Among them, $I_{u,v}$ represents the collection of items that all user v recommends to the user u, and the $\left|I_{u,v}\right|$ represents the collection and the total number of items in the $I_{u,v}$. The range of $trust(u,v)$ is between [0,1], and the higher the value of $trust(u,v)$ is, the higher the prediction accuracy of user v is. The higher the degree of trust of user u is, the higher the trust degree of user v will be. With the increase of the actual number of target users, the accuracy of the corresponding trust calculation will also be improved.

B. User Similarity Calculation

My algorithm needs to compute user preference and trust double information, so for each user, the similarity is decomposed into user preference similarity and user trust.

Based on the feature distribution of user preferences, the preference similarity between any two users u and v is calculated using cosine similarity[11] $usim(u,v)$, as shown in formula (8):

$$usim(u,v) = \frac{\sum_{j=1}^{k} p(u,j) \times p(v,j)}{\sqrt{\sum_{j=1}^{k} p(u,j)^2} \times \sqrt{\sum_{j=1}^{k} p(v,j)^2}} \tag{8}$$

Among them, $p(u,j)$ and $p(v,j)$ respectively represent the user preference distribution of user u and v on the j items. Here, we think that two users are fond of or disgusted with an object at the same time, all of them belong to a representation of user similarity.

Finally, using parameters α to balance the importance of user preference similarity and trust degree, the user similarity $sim(u,v)$ is calculated. For any two users u and V, the user similarity $sim(u,v)$ is shown by the formula (9):

$$sim(u,v) = \alpha \times usim(u,v) + (1-\alpha) \times trust(u,v) \tag{9}$$

When α=1, the user similarity degenerates to the user's preference similarity; when α=0, the user similarity degenerates to the user's trust degree; When α is (0,1), it is calculated on the basis of user preference and user trust.

The recommendation system goal is to different user preferences based on the personalized recommendation items, my paper calculates the user preferences, finally introduces user trust degree, comprehensive evaluation of multiple levels of similarity between users, to establish a more accurate distribution of user preferences, so as to realize the high quality personalized recommendation.

C. Time Based Weight of Interest

Each element in the evaluation matrix of the traditional collaborative filtering recommendation algorithm is a binary element of 0 and 1. In this case, each evaluation value is equal to the user, and the evaluation is static as time goes on. This is obviously not in accordance with the actual situation. In general, people are interested in different sources of interest and will change over time. The resources that users have visited recently are very useful to predict future trends of interest for users.

The resources that users have visited recently are very useful to predict future trends of interest for users. Inspired by the forgetting law, refer to the characteristics of the Ebbinghaus forgetting curve function, and set up $s(u,i)$ as the interest degree of the user u to the project i. Considering the time sequence of users' evaluation of all projects, t_0 is the earliest time for all scoring projects of user u, and t_i is the evaluation time for users to i. Then $s(u,i)$ can be expressed as:

$$s(u,i)=\frac{e^{-(t_i-t_0)}}{t_i-t_0} \tag{10}$$

If $t_0=t_i$, then $s(u,i)=1$ is defined.

Each user's interest change speed and rule are different, and user interest is also repeated. The above formula highlights the user's recent interest but neglects the impact of user's early access project on recommendation. Next, we introduce a metric $l(u,i)$ with long-term user interest. Set the set of items that the user u has already accessed to I_u. A time window T is defined, the collection of items that user u has visited in the recent T time period is l_{uT}, and l_{uT} reflects the user's recent interest to a certain extent. For a project, if u has access to recent project collections, many projects in l_{uT} have high similarity with i. It shows that project i is closely related to users' current interest, and users' interest may also be similar to project i in the future. So, project i plays a key role in predicting user interest.

Calculate $l(u,i)$ by the overall similarity of the project in i and l_{uT}:

$$l(u,i)=\frac{\sum_{j\in I_{uT}} sim(i,j)}{size(I_{uT})} \tag{11}$$

In this, $size(I_{uT})$ is the size of the resource collection that the user u has visited in the recent T time period.

From the above analysis, the recent interest measure function of user interest can be timely based on user interest, for user interest change frequently, and long-term user interest based on metric function is to make up for the missing key features of the recent user interest early data, unable to grasp the user interest are repeated, therefore, will measure function with long-term user interest measure function of user interest recently combined to get:

$$f(u,i)=\beta\times s(u,i)+(1-\beta)\times l(u,i) \tag{12}$$

Among them, β is a weight factor.

D. *Improved Collaborative Filtering Recommendation Algorithm*

We introduce the user interest of preference degree and time weight to the traditional collaborative filtering recommendation algorithm which only depends on user ratings, and propose an improved collaborative filtering recommendation algorithm.

Algorithm Collaborative filtering recommendation algorithm for user interest and relationship based on score matrix

Input User rating information, project score matrix R, project attribute matrix S, neighbor number K

Output Target user's prediction score

The concrete steps are as follows:

a) Calculate the user's attribute preference.

b) Calculate user attribute preference and the degree of user trust.

c) The preference of a user's project is calculated according to the value of step (a) and step (b).

d) The average score of the user and the sum of the preference are used as the preference value of the user to the project.

e) The original matrix is filled with the preference value calculated by the formula (4).

f) The use formula (9) is used to calculate the user's weight of interest based on time.

g) The formula(13) for calculating the similarity of the project is obtained by the weight of the step (6). The nearest neighbor of the target user is solved according to the result of the similarity calculation. For any item i, the similarity of target users is calculated according to formula (10), and the former K items with larger similarity are used as the nearest neighbor set of target items.

$$simt(i,j)=\frac{\sum_{u\in R(i)\cap R(j)}(f(u,i)\times(r_{uj}-\bar{r}_j))\times(f(u,j)\times(r_{uj}-\bar{r}_j))}{\sqrt{\sum_{u\in R(i)\cap R(j)}(f(u,i)\times(r_{uj}\times\bar{r}_j))^2}\times\sqrt{\sum_{u\in R(i)\cap R(j)}(f(u,j)\times(r_{uj}\times\bar{r}_j))^2}} \tag{13}$$

h) The user preference degree is added to the project score prediction formula. As shown in formula (14), we predict that the top K items with the highest score will be the top-N recommendation set of target user T.

$$\hat{r}_{ui}=\frac{\sum_{j\in R_i^K(u)} p(u,j)\times simt(i,j)\times r_{uj}}{\sum_{j\in R_i^K(u)} \left|simt(i,j)\times p(u,j)\right|} \tag{14}$$

IV. Experimental Results and Analysis

A) The experimental results and analysis are taken as an example of the hetcrec-2011-Last.fm public data set. The Last.fm dataset contains information about user's scoring information, user relationship, user's definition of item, tag's details, and user's time to project marking.

B) Standard of experimental evaluation

The target user u's prediction score for project i is expressed by $\hat{r}_{ui}$, and r_{ui} represents the true score of the test set. The Mean Absolute Difference (MAE) is used to evaluate the recommendation quality of the algorithm. The accuracy of the recommendation is calculated by calculating the deviation between the target user's prediction score and the target user's actual score. The lower the MAE, the more accurate the recommended method is. The MAE calculation is as follows:

$$MAE = \frac{\sum_{i}^{N} \left| \hat{r}_{ui} - r_{ui} \right|}{N} \tag{15}$$

Experiment I.

The effect of the ratio factor β on the time based interest is contrasted to the recommended algorithm. Because the range of β is [0, 1], the beta set from 0.1 - 0.9 in my paper. The existing research results also show that the nearest neighbor K will have a good recommendation when it is [30, 60][6].Therefore, calculate the change trend of MAE when K = 50. As shown in Figure 2, the different proportion factor β has a great effect on the recommended quality. Therefore, the importance of some data features can be taken into account by determining the importance of short term user access to the project by determining β.

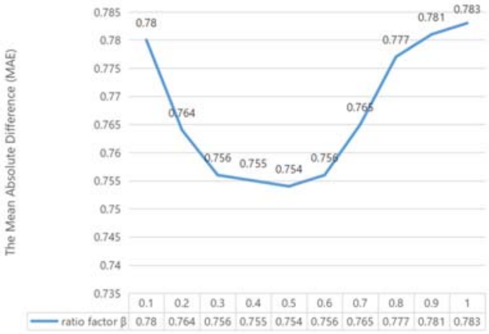

FIGURE II. MAE OF DIFFERENT RATIO B

Experiment II.

The MAE value of the collaborative filtering recommendation algorithm based on score matrix filling and user interest (PI-CF)is calculated in my paper. According to the optimal value set β=0.5 in Experiment 1, the MAE value obtained by experiment is compared with the traditional collaborative filtering recommendation algorithm based on user collaborative filtering and the collaborative filtering recommendation algorithm proposed in [5].The results of the experiment are shown in Figure III.

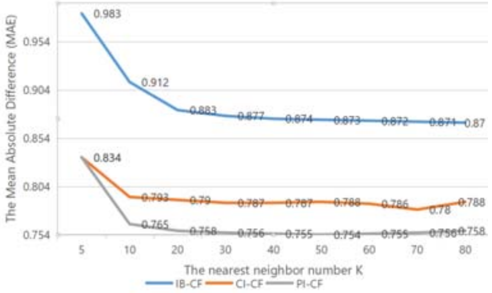

FIGURE III. MAE OF DIFFERENT NEIGHBOR NUMBER K

From Figure III, we can see that the user preference value is introduced into the user rating matrix, and the user's non scoring items are filled to solve the sparsity and improve the recommendation accuracy. And time based interest weight can highlight the importance of recent data, and avoid early data being ignored, which can more accurately reflect the trend of user interest change, and it can effectively improve the accuracy of recommendation.

V. CONCLUSION

Aiming at the characteristics of the traditional collaborative filtering recommendation algorithm, such as sparse data and no user interest, this paper solves the problem of sparsity to a certain extent, based on the preference degree of users to an item, and obtaining preference values before filling up the scoring matrix. At the same time, considering the user interest characteristics under time weighting, and comparing with other collaborative filtering recommendation algorithms, the results show that our algorithm can effectively improve the recommendation sparsity and improve the quality of recommendation system. In the next step, the collaborative filtering recommendation algorithm can be transplanted into the cloud computing environment to improve the parallelism and extensibility of the algorithm.

REFERENCES

[1] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In <em>Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining</em> (KDD '08). ACM, New York, NY, USA, 426-434. DOI: https://doi.org/10.1145/1401890.1401944

[2] Luo Xin,Liu Huijun,Gou Gaopeng,et al.A Parallel Matrix Factorization Based Recommender by Alternating Stochastic Gradient Decent[J]. Engineering Applications of Artificial Intelligence,2012,25(5):1403-1412.

[3] Wang Jun,de Vries A P,Reinders M J T.Unifying User-based and Item-based Collaborative Filtering Approaches by Similarity Fusion［C］/ /Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.New York,USA: ACM Press,2006: 501-508.

[4] Baralis E,Garza P.Item Selection for AssociativeClassification［J］.International Journal of Intelligent Systems,2012,27(3):279-299.

[5] Sarwar B,Karypis G,Konstan J,et al.Item-based collaborative filtering recommendation algorithms[C]// Proc of International Conference on World Wide Web.ACM,2001:285-295.