Research and Application of Algorithm Based on Maximum Expectation and Collaborative Filtering In Recommended System

Ying Fan1*, Huan Ma1, Zhongyang Chen1, Keli Shen1

1State Grid Shanghai Electric Power Research Institute, Shanghai 200438, China
*e-mail: fan_ying@sh.sgcc.com.cn

Abstract: The problem of sparse information is easily found in information searching for new users in recommended system, and it also has difficulty in recommending related modules. In view of the problem, the maximum expectation algorithm in demographics is adopted to cluster users for neighbouring users, and then it is regarded as an input of collaborative filtering algorithm. As the users’ scores on different projects represent their preference, certain demand relevance exists in similar users when evaluating the same project. And this kind of relevance degree varies with the change of individual demand. Therefore, a cooperative filtering algorithm based on the change of user’s demand is put forward by introducing a time weight function, which alleviates the shortness of traditional cooperative filtering recommendation algorithm. By tracking the needs of users, the scoring matrix can be predicted. According to experiments comparison, this algorithm can help solve the problem of sparseness of user’s scoring matrix and the recommendation quality is improved.

1.Introduction

1.1.Research status

In face of massive data, the demand of information for users can be easily satisfied. But with a huge increase in the amount of information over network, getting useful information is becoming harder and harder. Therefore, information explosion is formed and data mining website comes into being. Different kinds of recommendation systems using Collaborative filtering technology have been widely accepted with excellent application experience[1-5].

Collaborative filtering system [6] and Amazon personalized system [7] are two famous applications abroad. Tapestry [6] is the earliest collaborative filtering system which aims to solve the problem of information overload. To get the data set of the most similar neighbour of the target user, Group lens calculate user similarity through user-item scoring matrix and finally produce recommendations according to this data set [8]. By recommending personalized books to different users, Amazon personalized system helps increase sales greatly.

In China, Similar applications also turned up in recent year, with a significant increase in research level and citation scale. Among them there are two websites which perform better. They are “http://www.china-pub.com/” and “http://www.china-pub.com/” [9]. In the website of china-pub, members can get personalized customization according to their preference in the page named “my China-pub” which records the browsing history of users. The other website 360doc provides many related articles by establishing associations between highly relevant articles through article analysis.
1.2. Current problem

Before getting personalized services, the basic information of the user has already been recorded most of the time. But to new users, there is no browsing history which can be referred to, so it’s hard to get their demand though such sparse data. To solve this problem, the maximum expectation algorithm of demography is used to cluster new user’s basic information data and find its neighbouring user before using it as the input for collaborative filtering.

In the recommended system using collaborative filtering, data mining based on user’s demand is a complex problem [10-11]. User’s demand represents the preference of individual when making a choice among different recommended item. But the preference may change with user’s age, season and environment, etc. So, in this paper, the collaborative filtering algorithm based on user’s demand is proposed to describe the change of user’s demand and predict their scoring matrix.

2. The maximum expectation algorithm

2.1. Clustering analysis based on user’s feature

The problem of sparse information is easily found in the information search for new user in recommendation system, and the difficulty is produced when recommending the relevant module for users. In view of the problem, the maximum expectation algorithm in demographics is adopted to cluster users for neighbouring users [12], and then it is regarded as input of collaborative filtering algorithm [13-16].

When clustering users, gender, age, position and education are usually used as feature dimension. Because these feature information can have a great impact on user’s preference according to previous research.

2.2. The maximum expectation clustering algorithm

Even all the user data expressed as a are provided, it is still hard to determine which category the user belongs to. We use (a, b) represent all the user data, in which b represents the label of the branch to which A belongs with a range from 1 to θ. Then the probability density [17] of all the data can be defined as:

\[ f(a, b; \theta) = \sum_{i} r_i f_i(a, b; \theta) \]  \hspace{1cm} (1)

In equation (1), θ represents the number of density branches. \( r_1, r_2, ..., r_\theta \) represents the proportion of branches that account for the total branch. And \( f_i \) is density of the i-th branch.

\( \theta_i \) is the parameter of the corresponding branch to be determined. After the user data set \( \{x_1, x_2, ..., x_n\} \) is determined, the maximum likelihood function [18] can be used to calculate \( \theta_{\text{max}} \):

\[ \theta_{\text{max}} = \arg \max \prod_{i=1}^{n} f(a_i, b_i; \theta) \]  \hspace{1cm} (2)

The maximum expectation algorithm [19-21] is essentially an iterative algorithm, which can iterate from the initial solution \( \theta_0 \) then successively yield \( \theta_1, \theta_2, ..., \theta_t \). During the iterative process, the value of the likelihood function is always incremented. The algorithm flow is as follows:

1. \( \theta_0 \) is given as an initial distribution parameter.
2. Repeat the following process for each incremental parameter \( \theta_1, \theta_2, ..., \theta_t \) until converging.

Calculate the expectation of the Log likelihood function according to given user data and the current solution \( \theta_t \):

\[ Q(\theta \mid \partial_t) = \sum_{i=1}^{n} E_b[\log f(a, b; \theta) \mid \partial_i, \partial_t] \]  \hspace{1cm} (3)

In equation (3), \( E_b \) is the expectation of the random variable b.

Find a new parameter \( \partial_{t+1} \), which can make the expectation of the Logarithm likelihood function reaches the maximum:

\[ \partial_{t+1} = \arg \max Q(\theta \mid \partial_t) \]  \hspace{1cm} (4)
(3) $\partial_t$ can be finally obtained through loop iteration till algorithm convergence. Then various user clusters and distribution characteristics of each category can be obtained through the calculation of adaptive the maximum expected iteration.

3. The collaborative filtering algorithm

3.1. User-item scoring matrix

In recommendation system [22], there is a user-project scoring matrix between the user collection and the project collection, which can be expressed as table 1.

| Table 1. User-item scoring matrix $R_{sxt}$ |
|-------------------------------------------|
| $I_1$ | ... | $I_j$ | ... | $I_t$ |
| $U_1$ | $R_{1,1}$ | ... | $R_{1,j}$ | ... | $R_{1,t}$ |
| ... | ... | ... | ... | ... | ... |
| $U_i$ | $R_{i,1}$ | ... | $R_{i,j}$ | ... | $R_{i,t}$ |
| ... | ... | ... | ... | ... | ... |
| $U_s$ | $R_{s,1}$ | ... | $R_{s,j}$ | ... | $R_{s,t}$ |

In table 1, the user collection is expressed as $U=\{U_1, U_2, \ldots, U_s\}$ and the project collection is expressed as $I=\{I_1, I_2, \ldots, I_t\}$. The number of rows in the matrix is $s$ which means there are $s$ people. As the same, $t$ columns represent $t$ project. $R_{ij}$ represents the score of user $U_i$ on project $I_j$, which shows the preference of $U_i$. Since it’s impossible to get all the score of user on each item, there will be some blank column, resulting in the sparsity of the user-item scoring matrix. So the similarity calculation between different users becomes difficult, too.

3.2. Similarity calculation between users

3.2.1. Calculation using Standard cosine similarity

$$\text{Sim}(U_a, U_b) = \cos(R_a, R_b) = \frac{\sum_{k=1}^{s} R_{a,k} \times R_{b,k}}{\sqrt{\sum_{k=1}^{s} (R_{a,k})^2 \times \sum_{k=1}^{s} (R_{b,k})^2}}$$ (5)

The standard cosine similarity algorithm distinguishes vectors mostly from their direction. And it is insensitive to the numerical gap in each dimension, which may cause big mistakes. So adjustment must be taken by subtracting an average from the values of all dimensions.

3.2.2. Adjusting cosine similarity. In order to eliminate the scoring deviation of the same project by different users, and reduce the result bias, a certain average is subtracted from all dimensions.

$$\text{Sim}(U_a, U_b) = \frac{\sum_{l \in I_{ab}} (R_{a,l} - \overline{R}_a) (R_{b,l} - \overline{R}_b)}{\sqrt{\sum_{l \in I_{ab}} (R_{a,l} - \overline{R}_a)^2 \times \sum_{l \in I_{ab}} (R_{b,l} - \overline{R}_b)^2}}$$ (6)

In equation (6), $\overline{R}_a$ represents the average score of user $a$ on the same kind of projects. $R_{aj}$ represents the score of user $a$ on project $j$. $I_{ab}$ represents the collection of projects evaluated by both user $a$ and user $b$. $\text{Sim}(U_a, U_b)$ represents the similarity weight of user $a$ and user $b$. The value of $\text{Sim}(U_a, U_b)$ is between 0 and 1. The larger the value is, the more similar user $a$ and user $b$ are.

3.3. Improved algorithm considering the change of users’ demand

3.3.1. Weight function based on time. As shown in equation (7), the weight function based on time is proposed to have a better view in demands.
In equation (7), $t_{ui}$ indicates the time difference between the current time when the user accessing project $i$ and the last time when the user accessed project $i$. $L_{tui}$ indicates the time difference between the current time when the user accessing project $i$ and the first time when the user accessed project $i$. $\varepsilon$ is the weight change index between 0 and 1. The larger $\varepsilon$ is, the more often user access.

3.3.2 Improved cosine similarity. When calculating users’ similarity, the time-based weight function is multiplied by every score value in the user-item scoring matrix for higher precision.

$$
\text{Opt}_\text{sim}(U_a, U_b) = \frac{\sum_{i \in I} (R_{ai} \cdot \text{TWF}(u,i) - R_{ai}) (R_{bi} \cdot \text{TWF}(u,i) - R_{bi})}{\sqrt{\sum_{i \in I} (R_{ai} \cdot \text{TWF}(u,i) - R_{ai})^2 \cdot \sqrt{\sum_{i \in I} (R_{bi} \cdot \text{TWF}(u,i) - R_{bi})^2}}} \quad (8)
$$

3.4. The prediction of users’ score on certain project

After getting a more accurate value of the similarity between users, the blank of the User-item scoring matrix can be predicted. And the Global numerical algorithm is adopted by using the similarity of users' scores of similar products as weights.

$$
P_{rai} = R_{ai} + \left( \sum_{b=1}^{n} \text{Opt}_\text{sim}(U_a, U_b) \times (R_{bi} - \overline{R_{bi}}) \right) - 1 \quad (9)
$$

In equation (9), $n$ represents the number of users and $P_{rai}$ indicates the score predicted of user $a$ on project $i$.

4. The design and analysis of the proposed algorithm

4.1 The process of the proposed algorithm

input: feature dimension of users', user-item scoring matrix $R_{s \times t}$, the browsing history on $t$ project of $s$ user, and the weight change index $\varepsilon$.

Output: $N$ projects recommended to certain user

Step 1: according to given user set $U = \{U_1, U_2, \ldots, U_s\}$ and relevant feature data, Neighbouring user of the target user is to be found by maximum expected iterative clustering algorithm, as in equation (3) and (4).

Step 2: using user-item scoring matrix $R_{s \times t}$ and weight function based on time $\text{TWF}(u,i)$, the similarity of interest scoring of different users can be calculated, as in equation (8).

Step 3: the user’s scores on certain projects are predicted using global numerical algorithm, as in equation (9).

4.2 The flow of the proposed algorithm

as shown in figure (1), after the user log in, whether the information data is sparse or not is firstly to be judged. If so, neighbouring user has to be found using maximum expected iterative clustering algorithm. Then the data of the neighbouring user is used as the input of collaborative filter algorithm. Finally, relevant projects are recommended to target users according to the scoring similarity based on users’ interests calculated by user-item scoring matrix and time weight function.
User log in

Whether user’s information is sparse or not

No

Yes

User clustering to find neighbouring user

Collaborative filtering and predict user’s score

Resource recommendation and generate list

Figure 1. The flow of the proposed algorithm

4.3 Experimental analysis

4.3.1. The criteria of evaluation. Different kinds of criteria are proposed by researchers to test the effectiveness of the recommendation system. These criteria can be roughly divided into two categories. One is used to test the accuracy of recommended results. The other is used to test the time and space complexity of the Algorithm. Among all these criteria, the mean absolute error (MAE) is chosen in this paper for it can illustrate the error of the predicted value more accurately. MAE is obtained by differential accumulation between the predicted value and the true value.

Assume that the predicted score value can be expressed as \( \{ p_1, p_2, \ldots, p_n \} \) and the true value is \( \{ q_1, q_2, \ldots, q_n \} \), then MAE could be calculated by equation (10).

\[
\text{MAE} = \frac{1}{n} \sum_{i=1}^{n} |p_i - q_i|
\]

In equation (10), the value of MAE is inversely proportional to the recommended accuracy.

4.3.2. Data source. The data set provided by website http://grouplens.org/datasets/movielens/ is selected as a sample. The scoring data of 1,000 users on 1,700 movies have been extracted as experimental data.

4.3.3. Experimental environment. The experimental environment is shown in table 2.

| Operating system | Processor | Memory | Development tools | Development language |
|-----------------|-----------|--------|-------------------|----------------------|
| Windows 7 Ultimate | i5-2540   | 4GB    | Visual Studio 2012 | C#                   |
4.3.4. **Experimental results.** The final experimental results are shown as figure (2).

![Figure 2. Experimental results](image)

According to the experimental results, collaborative filtering algorithm based on changes in users’ demands is more accuracy than the traditional collaborative filtering algorithm.

5. **Application of the proposed algorithm**

The algorithm based on maximum expected clustering and collaborative filtering considering uses’ demand is successfully applied in the “power industry testing system”. A mobile vision of the testing system is developed using Android programming technology, which can provide personalized service for users. Its complete process is shown in figure 3.

![Figure 3. Application of the proposed algorithm](image)

6. **Conclusion**

In this paper, to solve the problem of sparse information of new users, the maximum expectation algorithm is adopted to cluster users. And to solve the problem of dynamic change of uses’ demands, weight function based on time is proposed to improve the accuracy of traditional collaborative filtering algorithms. In the future, we will continue to research related algorithms, improve their performance, and extend them to all walks of life.
References
[1] Tevreen, L., Hill, W., Amento, B., et al. (1997) A system for sharing recommendations. Communications of the ACM, 40 (3): 59-62.
[2] Shadanand, U., Maes, R. (1995) Algorithms for automating “word of mouth”. In: Proceedings of the computer-humaninteraction conference. Denver. pp.125-132.
[3] Rucker, J., Polanco, M. J. (1997) Personalized navigation for the Web. Communications of the ACM, 40 (3):73-76.
[4] Jian, S. (2004) Research on the recommendation of content-based e-learning guides. Ming Chuan University, Taiwan.
[5] Lieberman, H. Dyke N., Vivaqua, A. S. (1999) A collaborative web browsing agent In: International conference on intelligent user interfaces. pp. 67-75
[6] Sun, X. (2005) Study on sparsity and cold start of collaborative filtering system. : Zhejiang University. Hangzhou
[7] Yang, B., Zhao P. (2011) Overview of Recommendation Algorithms. Journal of Shanxi University: Natural Sciences Academic Edition, 34 (3): 337-350.
[8] Feng, Y. Research on collaborative filtering recommendation algorithm integrating user characteristics. (2014) Nanjing University of Posts and Telecommunications, Nanjing.
[9] Tu, J. (2013) Research on tag-based personalized recommendation technology. Chongqing University Science, Chongqing.
[10] Zhang, C., Chen, G., Wang, H. (2010) Recommendation model based on hybrid recommendation technology. Computer Engineering, 36 (22): 248-250.
[11] Liu, Z. (2015) Collaborative filtering recommendation algorithm based on user interests. International Journal of u-and e-Serv-ice, Science and Technology, 8(4): 311-320.
[12] Liu, G., Wang, M., Zhang, C., et al. (2011) Research on Data Mining in Mobile Learning. China Distance Education, 1: 31-35.
[13] Lu, D., Xu, J., Xiang, C., et al.(2016) Clustering method analysis in biological big dataAnalysis. Journal of Shanghai University: Natural Science Edition, 22 (1): 45-57.
[14] Xu, L. (2010) Algorithm and application of cluster analysis. Jilin University, Changchun.
[15] Chen, J., Wu, S., Sheng, C. (2008) Cluster analysis based on concept lattice. Shanghai University Journal: Natural Science Edition, 14 (4): 432-435.
[16] Du, R. (2012) Research on Bayesian classifier and its application. Shanghai University, Shanghai.
[17] Cheng, J., Xu, J. (2005) Supervised LVQ neural network based on EM algorithm and Its application. Systems Engineering and Electronics, 27 (1): 121-123.
[18] Li, H., Ma, X., Hu, Y., et al. (2014) A fusion of social networks and trust Research on sexual recommendation method. Journal of Computer Applications, 31 (3): 808-810.
[19] Huang, C., Yin, J., Wang, J., et al. (2010) Collaborative filtering recommendation with uncertain neighbors Algorithm. Chinese Journal of Computers, 33 (8): 1369-1377.
[20] Chen, Z., Yan, J. (2016) Collaborative filtering recommendation algorithm based on sparse data preprocessing Method. Computer Technology and Development, 26 (7): 59-64.
[21] Gao, Q., He, J. (2016) Improved collaborative filtering recommendation algorithm for data sparseness Method. Computer Technology and Exhibition, 26 (3): 63-66.
[22] Zhang, M. (2013) Comparison of cosine distance, Euclidean distance, and Jaccard similarity measure Analysis. http://www.cnblogs.com/chaoisimple/archive/2013/06/28/3160839.html.
[23] Zhao, Q., Lu, K., Wang, B. (2013) A memory-based propagation protocol the same filtering recommendation algorithm. Chinese Journal of Computers, 36 (3): 671-676.
[24] Wu, X. (2011) Research and Application of Software Development Method Based on Android Platform. Beijing University of Posts and Telecommunications, Beijing.