A Clustering Algorithm Solution to the Collaborative Filtering

Yongli Yang [1, a], Fei Xue [2, b], Yongquan Cai [1, c] Zhenhu Ning [1, d,*] and Haifeng Liu [3, e]

[1]Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China;

[2]School of Information, Beijing Wuzi University, Beijing 101149, China;

[3]Science and Technology on Information Systems, Engineering Laboratory, Beijing Institute of Control and Electronic Technology, Beijing 100038, China.

[a]yyyyll1218@163.com, [b]xuefei2004@126.com, [c]cyq940218@163.com, [d]nzh41034@163.com,

[e] haifeng4123@sina.com

Abstract

The recommendation system is widely used as a means of making effective use of large data and is widely followed by the people. Collaborative filtering recommendation algorithm cannot avoid the bottleneck of computing performance problems in the recommendation process. In this paper, we propose a collaborative filtering recommendation algorithm RLPSO_KM_CF. Firstly, the RLPSO (Reverse-learning and local-learning PSO) algorithm is used to find the optimal solution of particle swarm and output the optimized clustering center. Then, the RLPSO_KM algorithm is used to cluster the user information. Finally, give the target user an effective recommendation by combining the traditional user-based collaborative filtering algorithm with the RLPSO_KM clustering algorithm. The experimental results show that the RLPSO_KM_CF algorithm has a significant improvement in the recommendation accuracy and has a higher stability.

Keywords

Collaborative Filtering Recommendation Algorithm;RLPSO Algorithm;K-means Algorithm.

1. Introduction

The recommendation system played an important role in the video, news, social network, music, books, electricity business and other fields as a way to make effective use of large data with the rapid development of information technology [1]. In terms of collaborative filtering, it can be divided into user-based and item-based recommendations. Machine Learning Model that concluded LFM, ALS, Limited Boltzmann Machine[2] and a series of model-based recommendation algorithm is also increasing in the development of artificial intelligence today[3].

However, despite the recommendation system have attracted much attention in the enterprise and the Internet,there are other issues like cold start, sparseness and for ZB-level data on how to quickly deal with in the recommendation process. The user and project information are clustered to form several user-project subgroups and the experiment shows that the accuracy of the proposed algorithm is improved compared with the original algorithm [4,5]. The authors in [6] propose the algorithm which accurately identifies the user's personal interest and effectively improves the recommendation accuracy based on the combination of temporal behavior and probability matrix decomposition. The hierarchical weighted similarity is introduced to measure the similarity of users at different levels in order to select the neighboring users of the target that can significantly improve the scoring effect [7].

The authors in [8] proposes the calculation of the similarity of mobile users across the project using the distance of pushing machine and the algorithm alleviates the influence of scoring data sparse on the collaborative filtering algorithm and improves the recommendation accuracy.Faced with these problems that processing of data in the recommendation system and the bottleneck problem of computing speed , the collaborative filtering recommendation algorithm user's neighbor refers to all

users. However, users with higher similarity are clearly more valuable than other users. So this paper proposes RLPSO_KM_CF collaborative filtering recommendation algorithm.

2. Related Works

2.1 Traditional User-based Collaborative Filtering Algorithm

The traditional User-CF collaborative filtering algorithm uses the target user's preference information to compute the neighborhood user set similar to the target user and then recommend the valid item to the target user [11]. This paper uses the Person correlation coefficient to calculate the correlation between users. The user similarity formula is as follows:

$$sim(u_i,u_j)=\frac{\sum_{i_c\in I_{i,j}}\left(r_{u_i,i_c}-\overline{r_{u_i}}\right)\left(r_{u_j,i_c}-\overline{r_{u_j}}\right)}{\sqrt{\sum_{i_c\in I_{i,j}}\left(r_{u_i,i_c}-\overline{r_{u_i}}\right)^2}\sqrt{\sum_{i_c\in I_{i,j}}\left(r_{u_j,i_c}-\overline{r_{u_j}}\right)^2}} \tag{1}$$

formula 1, $\overline{r_{u_i}}$ and $\overline{r_{u_j}}$ are the average ratings to user u_i and u_j, r_{u_i,i_c} and r_{u_j,i_c} are the ratings for item i_c to user u_i and u_j. Define the prediction ratings formula as follows:

$$R(u_i,i_i)=\overline{r_{u_i}}+\frac{\sum_{u_j\in N_{ui}} sim(u_i,u_j)(r_{u_j,i_i}-\overline{r_{u_j}})}{\sum_{u_j\in N_{ui}} sim(u_i,u_j)} \tag{2}$$

formula 2, $\overline{r_{u_i}}$ and $\overline{r_{u_j}}$ are described in formula 1, r_{u_j,i_i} is ratings for item i_i to user u_j, N_{u_i} is neighborhood collection to user u_i.

2.2 RLPSO Optimization Algorithm

The RLPSO algorithm is an improved PSO algorithm [9]. The algorithm performs local search by the difference of the historical position of the particle swarm. At the same time, the algorithm introduces the inverse learning sub-particle swarm in order to avoid the premature convergence [10].

2.3 K-means Algorithm

Clustering algorithms are followed in the field of data mining and artificial intelligence, K-means algorithm is also popular, which the input value is the number of clustering k and n data objects used, the output value is k clustering Datasets[11].

3. RLPSO_KM_CF Algorithm

This section will describe the RLPSO_KM_CF algorithm in detail. Firstly, it describes how to improve the K-means clustering algorithm. Then, the application of RLPSO_KM algorithm in collaborative filtering algorithm is expounded.

3.1 RLPSO_KM Algorithm Based On RLPSO

RLPSO_KM algorithm is described as follows:

Input: the Datasets D, the cluster number k, the particle swarm size N, the reverse learning particle swarm size n, the particle swarm learning factors c_1 and c_2, the reverse learning factors c_3 and c_4, the maximum iteration number of the particle swarm, the reverse learning iteration times L_{times}, the maximum inertia weight ω_{max}, the minimum inertia weight ω_{min}, the disturbance coefficient d_0, the time factor H_0, the maximum particle flying velocity v_{max}.

Output: Optimized k clustering centers.

Step 1: Initialize the particle swarm. From the Datasets D randomly selected k data items as the particle position and velocity of each dimension of the initial value and loop this process N times;

Step 2: Initialize the particle swarm optimal position and suboptimal position. Calculate the fitness value of each particle in the particle group by using fitness formula to select the initial value of the optimal and suboptimal position of the particle population;

Step 3: Initialize the worst particle swarm W;

Step 4: Iterate search for particles;

While (t< tmax ||ρ<10e-6)

A. Adjust ω according to the weight adjustment formula;

B. Update the particle position and velocity under the position and speed update formula;

C. Calculate f(x) for each particle in the light of the fitness formula;

D. Update the optimal particle value;

E. Update Pg1 and Pg2;

F. Local search under the search formula ;

G. Adjust d0 in line with the perturbation coefficient formula;

H. If meet the reverse learning conditions (the algorithm local convergences or reaches the thresholds) adjust the vmax;

H1. Update the speed and position of the reverse learning particle according to the reverse learning speed and position formula;

H2.Update the position and velocity of the remaining particles in reverse learning according to the position and speed update formula of the reverse learning;

End If

I. Calculate ρ according to convergence function ;

J. if (ρ> thresholds)

break;

K.t ++;

End While

Step 5: Output the optimal solution of the particle swarm;

Step 6: Run the K-means clustering algorithm and output the optimized clustering centers;

End

3.2 RLPSO_KM_CF Algorithm Based On RLPSO_KM

Users with higher similarity to the target user have a more valuable reference than other users. The RLPSO_KM clustering algorithm is used to cluster the user information and then the target user is effectively recommended by using the traditional user-based collaborative filtering algorithm each cluster. And recommend the most popular items to the new target users. The formula of the item popularity is as follows:

$$ItemPop_i = \frac{|U_i|}{\sqrt{\sum_{i \in I} |U_i|^2}} \tag{3}$$

RLPSO_KM_CF algorithm is described below:

Input:cluster number k, iteration times m. ratings information, recommended number of the items N.

Output: Top-N recommendation.

Begin

Step 1:If（Whether the target user is a new user）

A.Calculate ItemPopi under the formula 3 to form the collection W;

B. Descending Sort W to form Wnew;

C. Select the top N popularity from the Wnew to form Target;

D. Recommend item to the target user;

End If

Step 2: Calculate the cluster center under RLPSO_KM algorithm;

Step 3: Calculate the cluster to which the target user belongs by the formula 1;

Step 4: Using the traditional collaborative filtering algorithm for the target user to recommend in the cluster;

Step 5: Output Top-N Recommended List;

End

4. Experiments

4.1 Experimental Environment

The experimental use the centos7.0 device system server, which contains seven work nodes and a master node. Spark version is 2.0, Hadoop version is 2.7. This paper uses the University of Minnesota Movie Lens as experimental data. In this paper, three methods are selected as the contrast algorithm: the traditional UserCF collaborative filtering recommendation algorithm, the improved Top-N clustering collaborative filtering recommendation algorithm KCF, and the RLPSO_KM_CF algorithm.

4.2 Experimental Results

In this paper, we use the recall rate and MAE to evaluate the experimental results. In Fig 1, the MAE curve is drawn under the MovieLens1M datasets. It can be clearly seen that the MAE value of the RLPSO_KM_CF algorithm is the fastest when the clustering factor increases at the beginning of the experiment. When the clustering factor is 4, the RLPSO_KM_CF MAE value is the smallest and the result is best. The MAE value tends to increase first and then decrease when the clustering factor increases.

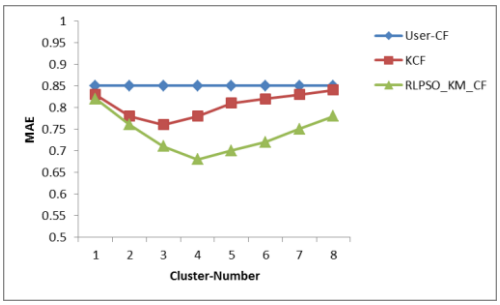

Fig.1 Based on the MoviesLens1M Datasets

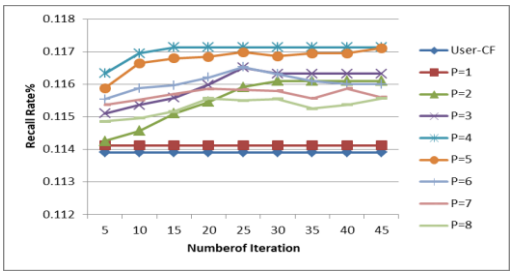

Fig.2 Recall Rate (Different iterations)

Fig 2 is the recall rate of the RLPSO_KM_CF algorithm under different iterations. The abscissa represents the number of iterations of the clustering algorithm and the ordinate indicates the recall rate of the recommended results. When the iterations are about 25, the recall rate basically has achieved the maximum. When the clustering factor k is 4 and the iterations are about 15, the algorithm is obviously convergent, and the recall rate is 0.117136. Compared with the traditional collaborative filtering algorithm, RLPSO_KM_CF algorithm is improved by 3.2%, which is 1.1% higher than the KCF algorithm. It also confirms that the target user's neighborhood set is relatively small and the recommendation accuracy will be reduced with the clustering factor increasing.

5. Conclusion

In the traditional collaborative filtering recommendation algorithm user's neighbor refers to all users. However, users with higher similarity are clearly more valuable than other users. This paper proposes a collaborative filtering algorithm RLPSO_KM_CF.The RLPSO_KM algorithm is used to cluster the user information, and the traditional collaborative filtering algorithm is combined with the RLPSO_KM cluster to effectively recommend the target user. We can consider choosing some clustering algorithms suitable for sparse matrix in the future research.

Acknowledgements

We would like to express sincerely our thanks to the teachers and students who have given support and advice on the work of this paper.

References

[1] Ricci F, Rokach L, Shapira B. Introduction to Recommender Systems Handbook[M]// Recommender Systems Handbook. Springer US, 2011:1-35.
[2] Salakhutdinov R, Mnih A, Hinton G. Restricted Boltzmann machines for collaborative filtering[C]// International Conference on Machine Learning. ACM, 2007:791-798.
[3] Zhen hua HUANG , Jia wen ZHANG, Chunqi TIAN , et al.Study on recommendation algorithm based on sorting learning [J] .Journal of Software, 2016, 27(3):691-713.
[4] Xu B, Bu J, Chen C, et al. An exploration of improving collaborative recommender systems via user-item subgroups[C]// 2012:21-30.
[5] Chen Z, Cai D, Han J, et al. Locally Discriminative Coclustering[J]. IEEE Transactions on Knowledge & Data Engineering, 2012, 24(6):1025-1035.
[6] Guangfu SUN, Le WU, Qi LIU, et al. Cooperative filtering recommendation algorithm Based on timing behavior [J].Journal of Software, 2013(11):2721-2733.
[7] Wenqiang Li,HongJi Xu,Mingyang Ji,Zhengzheng Xu,Haiteng Fang.A Hierachy Weighting Similarity Measure to Improve User-Based Collaborative Filtering Algorithm[C].2016 2nd IEEE International Conference on Computer and Communications.2016:843-846.
[8] Xun Hu,Xiangwu Meng,Yujie Zhang,et al. A Recommendation Algorithm for Converting Project Characteristics and Mobile User Trust Relationship [J].Journal of Software, 2014 (8): 1817-1830.
[9] Kenndy J,Eberhart R C,Particle swarm optimization//Proceedings of the IEEE International Conference on Neural Networks.Piscataway,USA,1995,4:1942-1948.
[10] Xuewen XIA, Jingnan LIU, Kefu GAO, et al.Particle swarm optimization with reverse learning and local learning ability [J].Journal of Computers, 2015(7):1397-1407.
[11] JiaWei Han Micheline Kamber Jian Pei.Data Mining Concepts and Techniques Thrid Edition[M].Machinery Industry Press,2012:293-297.