An Improved Collaborative Filtering Algorithm Based on RLPSO

YONGLI YANG, ZHENHU NING, YONGQUAN CAIC FEI XUE, SHIQIANG ZHANG and HAIFENG LIU

ABSTRACT

Confront of the large amount of data generated by the Internet and how to make the inherent advantages. The recommendation system is widely used as a means of making effective use of large data and is 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 an improved 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, the traditional collaborative filtering algorithm is combined with RLPSO_KM clustering to effectively recommend the target user. The experimental results show that the RLPSO_KM_CF algorithm has a significant improvement in the recommended accuracy and has a higher stability.

KEYWORDS
Collaborative Filtering Algorithm, RLPSO Algorithm, K-means Algorithm.

INTRODUCTION

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. But 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]. The authors in [5] propose the algorithm based on the combination of temporal behavior and probability matrix decomposition. The algorithm accurately identifies the user's personal interest and effectively improves the recommendation accuracy. 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 [6]. Through the time factor to make up for the user's preferences change, training model separate from prediction phase, and the time series of the user is applied to the prediction phase without increasing the complexity of the training model to improve the recommended accuracy[7]. The work in [8] defines the candidate neighbors through the development of distributed scoring management strategy and to ensure the recommendation mechanism to quickly locate candidate neighbors by building the index. So predict the target user's top-N recommendation set in the cloud computing environment. The results show that the algorithm improves the recommendation accuracy and effectiveness. Confront of these problems that processing of data in the recommendation system and the bottleneck problem of computing speed. At the same time, 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.

**RLPSO_KM_CF ALGORITHM BASED ON RLPSO**

The RLPSO algorithm is an improved PSO algorithm [9] which performs local search by the difference of the historical position of the particle swarm. And the algorithm introduces the inverse learning sub-particle swarm avoids the premature convergence [10].

Clustering algorithms are followed in the field of data mining and artificial intelligence, K-means algorithm [11] 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.

This paper first introduces the RLPSO_KM algorithm which is described below. Then the RLPSO_KM algorithm is used to cluster the user information. Then, the traditional cooperative filtering algorithm [12] is combined with the RLPSO_KM cluster to effectively recommend the target user.

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 c1 and c2, the reverse learning factors c3 and c4, the maximum iteration number of the particle swarm tax, the reverse learning iteration times Limes, the maximum inertia weight
mix, the minimum inertia weight man, the disturbance coefficient d0, the time factor H0, the maximum particle flying velocity vamp.

Output: Optimized k clustering centers.

Begin

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 vamp;

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

EXPERIMENTS AND RESULTS

The Experimental Environment and Parameter Settings

According to the research, 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. The RLPSO_KM clustering algorithm is used to divide the users into 2, 3, 4, 5, 6, 7 and 8 clusters, and the traditional user-based collaborative filtering recommendation algorithm is applied to each cluster in each time. Finally, calculate the average recall rate and MAE value.

Experiments Results

Table 1 is the recall rate of the RLPSO_KM_CF algorithm when the number of clusters is different. Among them, when the number of clusters is 1, RLPSO_KM_CF algorithm is the traditional UserCF algorithm. 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. But the algorithm recommended accuracy will decline when the number of users who calculate the similarity with the target user will decrease as the clustering factor increases.

<table>
<thead>
<tr>
<th></th>
<th>5</th>
<th>10</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
<th>35</th>
<th>40</th>
<th>45</th>
</tr>
</thead>
<tbody>
<tr>
<td>User-CF</td>
<td>0.113911</td>
<td>0.113911</td>
<td>0.113911</td>
<td>0.113911</td>
<td>0.113911</td>
<td>0.113911</td>
<td>0.113911</td>
<td>0.113911</td>
<td>0.113911</td>
</tr>
<tr>
<td>P=1</td>
<td>0.114125</td>
<td>0.114125</td>
<td>0.114125</td>
<td>0.114125</td>
<td>0.114125</td>
<td>0.114125</td>
<td>0.114125</td>
<td>0.114125</td>
<td>0.114125</td>
</tr>
<tr>
<td>P=2</td>
<td>0.114253</td>
<td>0.114568</td>
<td>0.115103</td>
<td>0.115458</td>
<td>0.115914</td>
<td>0.116102</td>
<td>0.116102</td>
<td>0.116102</td>
<td>0.116102</td>
</tr>
<tr>
<td>P=3</td>
<td>0.115106</td>
<td>0.115364</td>
<td>0.115587</td>
<td>0.115986</td>
<td>0.116513</td>
<td>0.116324</td>
<td>0.116324</td>
<td>0.116324</td>
<td>0.116324</td>
</tr>
<tr>
<td>P=4</td>
<td>0.116332</td>
<td>0.116946</td>
<td>0.117136</td>
<td>0.117136</td>
<td>0.117136</td>
<td>0.117136</td>
<td>0.117136</td>
<td>0.117136</td>
<td>0.117136</td>
</tr>
<tr>
<td>P=5</td>
<td>0.115876</td>
<td>0.116648</td>
<td>0.116796</td>
<td>0.116843</td>
<td>0.116984</td>
<td>0.116854</td>
<td>0.116948</td>
<td>0.116948</td>
<td>0.117106</td>
</tr>
<tr>
<td>P=6</td>
<td>0.115536</td>
<td>0.115874</td>
<td>0.115968</td>
<td>0.116205</td>
<td>0.116532</td>
<td>0.116314</td>
<td>0.116103</td>
<td>0.116002</td>
<td>0.116006</td>
</tr>
<tr>
<td>P=7</td>
<td>0.115369</td>
<td>0.115524</td>
<td>0.115698</td>
<td>0.115859</td>
<td>0.115836</td>
<td>0.115795</td>
<td>0.115568</td>
<td>0.115869</td>
<td>0.115597</td>
</tr>
<tr>
<td>P=8</td>
<td>0.114863</td>
<td>0.114962</td>
<td>0.115164</td>
<td>0.115568</td>
<td>0.115502</td>
<td>0.115548</td>
<td>0.115256</td>
<td>0.115368</td>
<td>0.115568</td>
</tr>
</tbody>
</table>

Figure 1. Based on the MoviesLens1M Datasets.
Fig 1 shows the MAE curve drawn under the 1M data set of the Movie Lens. 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 MAE value of RLPSO_KM_CF is the smallest, and the result is the best. With the clustering factor increases the MAE value gradually increases after decreasing from Fig 1. This also shows that the target user neighborhood set is relatively small and the recommendation precision is reduced with the increase of the clustering factor.

CONCLUSION

Users with higher similarity are clearly more valuable than other users. So this paper proposes RLPSO_KM_CF collaborative filtering recommendation algorithm. 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 and traditional cooperative filtering algorithm in the face of cold start problem and scarcity of scoring matrix in the future research.

ACKNOWLEDGEMENTS

Corresponding author: Zhenhu Ning, Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China zh41034@163.com.

REFERENCES