International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.4 (2015), pp.71-80
http://dx.doi.org/10.14257/ijmue.2015.10.4.08

The Research of the Recommendation Algorithm in Online Learning

Ruiguo Yu[1], Zhiyong Cai[2], Xiuping Du[3,*], Muwen He[4], Zan Wang[5], Binlan Yang[6] and Peng Chang[7]

[1,2,6]*School of Computer Science and Technology, Tianjin University, Tianjin, China*
[1,2]*Tianjin Key Laboratory of Cognitive Computing and Application*
[3]*School of Education, Tianjin University, Tianjin, China*
[5]*School of Computer and Software, Tianjin University, Tianjin, China*
[4,7]*Information and Network Center, Tianjin University, Tianjin, China*
rgyu,caizhiyong2014,duxiuping,hemuwen,wangzan,blyang,snow}@tju.edu.cn

Abstract

Recommendation algorithm is a kind of method in information filtering and has been widely applied on Internet. Collaborative filtering is widely used in the recommendation systems and has turned out to be successful. With the growth of the resources, it is difficult for users to find learning resources that suit for themselves. The recommendation algorithm is required to analyze the behavior of the users and then recommend object of learning. Online judge is a kind of the online learning. People can evaluate their programming ability through online judge. The performance of the different recommendation algorithms is analyzed in this paper and it is proposed that the item-based collaborative filtering recommendation algorithms should be applied into the online learning system. Based on the algorithm, we propose that by pre-processing the data and using user data that have solve large number of problems, we can get a better recommendation result.

Keywords: *Online Learning, Online Judge, Recommendation Algorithm, Collaborative Filtering*

1. Introduction

With the development of the Internet, the amount of information in the world is increasing far more quickly than our ability to process it. Recommendation algorithm, one of the information filtering method, can help us to filter the information we need. Meanwhile, compared with the traditional category filtering methods and search engine, the recommendation algorithm provides more personalized service. The recommendation algorithms have recently applied into the E-commerce. The Amazon has put the recommendation algorithm into practice and has made a great success at board [1]. The recommendation algorithms have also been used to recommend some other items, such as films [2].

Online learning has been emerged with the Internet. Users can learn their courses at anytime and anywhere through the Internet [3]. Each user can learn what they are interested in. The users can also use the online judge to evaluate their programming abilities.

In this study, we analyze the performance of the different recommendation algorithms from different fields, and propose that the item-based collaborative filtering recommendation algorithms should be applied into the online learning system. Based on this algorithm, we figure out that if we select the users who have solve more problems as

ISSN: 1975-0080 IJMUE

our training set and give the recommendation to every user, the personalized recommendation will be more successful.

2. Related Work

2.1. The Recommendation Algorithms

With the development of the Internet, the problem of information overload seems to crop up more and more often. The recommendation algorithm as a kind of method in information filtering is a great alternative. There are two general classes of recommendation algorithms: the collaborative filtering recommendation algorithms and content-based recommendation algorithms [4]. The collaborative filtering algorithm has been first proposed in the 1990s, and its core idea is to recommend items dependent on the records of other users [5].The collaborative filtering recommendation is one of the most widely applied at present [6].

2.2. The Collaborative Filtering Recommendation Algorithms

The CF is the process of filtering or evaluating items through the opinions of other people. After the concept of the CF was proposed, many recommendation algorithms based on the idea of the CF have been applied into the recommendation systems. For example, GroupLens makes use of the CF recommendation algorithm to recommend films [2]. Amazon has taken advantage of the CF recommendation algorithm to recommend items and makes a great success [8]. At present, CF recommendation algorithms are been classified into the memory-based CF recommendation algorithms and the model-based CF recommendation algorithms [9].The biggest advantage of the CF recommendation algorithms is that the algorithms can give the users recommendations and require fewer descriptions about the items. That means that the system can recommend the commodity, which is of difficulty to describe, such as films, books and music.

The memory CF recommendation algorithms contain the item-based CF recommendation algorithms and the user-based CF recommendation algorithms [9].The user-based CF recommendation algorithms pay attention to the similar users. They usually have two steps to decide which items should be recommended. These kinds of algorithms firstly identify the most similar users (nearest neighbors) to the active user. Then the algorithm will give the active user the recommendation list [7]. The item-based CF recommendation algorithms come from the similar idea. They firstly calculate the similarities between the different items. Then the algorithms will give the similar items to the items the active user has paid attention to [8].

The model-based CF algorithms learn to recognize complex patterns based on the training data, and then make intelligent predictions for the CF tasks. These kinds of the algorithms are usually dependent on the probability. The model-based CF recommendation algorithms, such as Bayesian models, clustering models, and dependency have been investigated [9].

The hybrid CF recommendation algorithms combine CF with other recommendation algorithm to make predictions or recommendations. In some conditions, the hybrid algorithms can achieve better results. Some hybrid CF algorithms, such as the content-boosted CF algorithm, are found helpful to address the sparse problem, in which external content information can be used to produce predictions for new users and items [11].

3. The Application of the Recommendation Algorithms in Online Learning

3.1. The Online Learning

The online learning refers to the use of the Internet for self-learning on online learning platform. Compared with the traditional approach to learning, the online learning has its own significant advantages. Firstly, because of the online learning, people learn more variously than before. Secondly, the online learning can support people learn at any place and any time through the Internet. Thirdly, the users are truly interested in learning by themselves. Finally, the users can exchange ideas on some issues on the platform.

However, the online learning has also some disadvantages. While learning on the platform, the users lack the guidance of teachers, which makes the knowledge a little scattered. If the situation lasts long, the users will waste their time and cannot get the results they expect. So we want to give users some recommendations in the online learning platform, in order to help users make full use of online learning resources, improve the learning efficiency and the users' level of knowledge.

3.2. The Description of the Problem

The Tianjin University Online Learning is an online learning platform. The users can solve the problems by programming, submit their own programs and judge the program whether it is right or wrong on this platform, which means users can test their own programming ability by this way. However, the number of problems is so large and their difficulties are quite various. In order to help the users take advantage of the system more effectively, the system should recommend the problem marching with their ability. That means that the users have more time to improve their ability than before, which may enhance the user experience.

Among the above, the recommendation algorithms should be applied into the system. But if we use the content-based recommendation algorithms, users will get the similar problems all the time. After a period of time, the users cannot get any new problems. The users cannot keep improving the ability at the same time.

So the CF recommendation algorithms should be considered. The user-based CF recommendation algorithm is a choice. However, when we calculate the neighbors, so many users lead to a great amount of time, which the users could not wait. The model-based CF recommendation algorithm sometimes makes use of some machine learning methods, such as the clustering methods, the Bayes' networks, but the online learning system has recorded nothing but the records that the users have solved problems. The item-based CF recommendation algorithm has its own advantages. We can assume that the users who have solved the same problems have the same ability. We recommend the similar problems, which the similarity is just dependent on the other users. One of advantages is that we only focus on the problems people have resolved, which means the number we count is much more less. Another advantage is that we can calculate the similarity in offline. Above all, we decide to use the item-based CF algorithms to give the users recommendation. We assume that a new user who first uses the online learning system doesn't have the strong programming skills. If we use the item-based CF recommendation algorithms, the problems the system recommend will not be too difficult and can march the users' ability properly.

3.3. The Application of the Item-based CF Recommendation Algorithms in Online Learning Platform

The item-based CF recommendation algorithm includes two steps. The first step is to calculate the similarity between the different items. In our study, we have made a

pretreatment before we calculate the similarity; we select the users who have solved many problems. Then we calculate the similarities between the different users. We will give the details of the algorithms.

(1) The pretreatment of the data and Initialization
   a. Pick up the problems the users have solved.
   b. Count the number of problems users have solved and pick up the users have solved more problems than the threshold we set.
   c. Set up a user-problem matrix *R*, *1* stands for the problem that the user have solved , and *0* stands for the problem the user haven't.

(2) Set up the similarity model.

The point of the recommendation algorithm is to set up the similarity model. The input of the algorithm is the matrix *R*, and the parameter *k*, and the output is the similarity matrix *M*. The parameter *k* is used to describe the *k* most similar problems. In the algorithm, we will calculate all the similarities between the problems and keep the k-most similar problems saved in the matrix M. The detail of the algorithm is described in the Table 1.

The Description of the Algorithm 1

Algorithm 1: Set up the similarity model

Input: *R*, *k*
Output: *M*
for i→1 to m do
  for j→ 1 to m do
    if i !=j then calculate sim(i,j)
    else sim(i,j)=0
   end if
end for
To the problem i, Rank other problems descending sort by the similarity with the problem i. pick up the first ***k*** similar problems and insert the problems information and the value of similarity into the matrix M.
end for
return M

(3) Recommend Problems:

After setting up the similarity model, we can give the users the recommendation. We use the m-dimensional column vector U to represent the problems the users have solved. *N* stands for the number of the recommended problems. The detail of the recommendation algorithm is described in the Table 2.

The point of the algorithm is how to calculate the similarities between different problems. The main methods to calculate the similarity are the vector cosine-based similarity, the adjusted cosine similarity and the Pearson similarity.

Formally, in the user-problem matrix, the cosine similarity between the problem *i* and the problem *j* denoted *sim(i.j)* is given by [12]

$$sim(i, j) = \cos(i, j) = \frac{\vec{i} \bullet \vec{j}}{\| \vec{i} \| \bullet \| \vec{j} \|} \quad (1)$$

where ● denotes the dot-product of the two vectors.

The Description of the Algorithm 2

Algorithm 2. Recommend problems

Input: M, U, N
Output: r
r=M*U
for i→1 to m do
if the problem has been solved by the user, the value in the vector r will be assigned to 0.
end for.
if the value in the vector R cannot be ranked at the top N, the value will also be assigned to 0
return r

The adjusted cosine-based similarity and the Pearson similarity will not be considered. Because both of the methods are needed to calculate the average score. In our study, we only record 1 or 0 as a result. The vector cosine-based similarity is a choice but it's not enough. Meanwhile, a question has been come up with. The *sim(i,j)* and *sim(j,i)* may not be different [9]. So the conditional probability is proposed to solve the problem by the Mukund and Ge-orge [9].The Bayes formula is given by the equation (2).

$$p(i \mid j) = \frac{p(i,j)}{p(j)} \tag{2}$$

The *p(i,j)* means the probability of the problems the user *i* and the use *j* have both solved. However, the similarity of the problem *i* with the problem *j* and the similarity of the problem j with the problem i is different. So we can improve the conditional probability to solve the different similarities between the two problems. And the kind of the similarity can be written as

$$sim(i,j) = \frac{p(i,j)}{p(i) \bullet p(j)^{\alpha}} \tag{3}$$

In the practical problem, the *p(i,j)* means the probability of the users who have both solved the problems *i* and the problem *j*. *p(i)* means the probability of the users who have solved the problem *i*. The $\propto$ is a parameter. If the $\propto$ is 0, the similarities of the both problems will be different. If the $\propto$ is 1, the similarities of the both problems will be the same.

4. The Experiment

In this section, we investigate which the parameter $\propto$ and number *k* are proper. And then we will compare the method of the cosine-based similarity to the method of the conditional probability. We propose that by preprocessing the data, using user data that have solve a large number of problems, we can get a better recommendation result.

4.1. Data Sets and Evaluation Metrics

The data sets we use the data of the online judge system of the Tianjin University. The data sets include all the records that all of the users have tried to solve the problem, which is solved and not solved. We only use the records that the users have solved the problems. We use the recall, which is often used in the CF recommendation algorithms. The recall is given by

$$recall = \frac{m}{n} \tag{4}$$

Where m is the number of the accurate predicted users, n is the total number of the users.

4.2. The Traditional Item-based Recommendation Algorithm

(1)The effect of the parameter $\propto$

We choose the conditional probability to set up the similarity model. By changing the parameter $\propto$, we can select the best parameter $\propto$. When we calculate the similarity, the para-meter $\propto$ varies from 0 to 1. In our study, we increase the parameter $\propto$ by 0.1 each time. We also investigate the effect of the different k. So we increase the k by 10 each time. And the Figure 1 shows the effect of the different k and the different $\propto$ and we use the recall to evaluate it. What we will see is that the best result occurs when the parameter $\propto$=0.8. The Figure 2 shows the effect of the k when we choose the parameter $\propto$=0.8.

4.3. The Item-based CF Recommendation Algorithm by Group Users

The performance of the traditional CF recommendation algorithm is not satisfactory. The reason might be the data sets of the sparseness. So we divide the users in-to groups and select users to set up the similarity model, in order to eliminate the effects of the sparseness.

According to the number of the problems the users have solved, we divide the users into two groups: the one is included the users who have solved problems less than 100 problems and the other is more than 100.We use the group whose users have solved more problems to set up the similarity model and then give the recommendation to the group 1 and group 2.

The Figure 3 shows that we make use of the conditional probability similarity to give the recommendation to the group 1.And we can see from the Figure 3 with the development of the parameter $\propto$, the recall also rises. But while the parameter $\propto$ equals 1, the declination of the recall is shape. When the parameter $\propto$ is equal to 0.9, the recall will get the best. Then we choose the parameter $\propto$ equals to 0.9 when giving re-commendation to the group 1 and the group 2. And the result shows in the Figure 4. The Figure 4 shows that the group 1 is much better than the group 2. The recall of the group 1 is nearly 0.25.And when the k varies from 10 to 30, both of the groups will get better results.

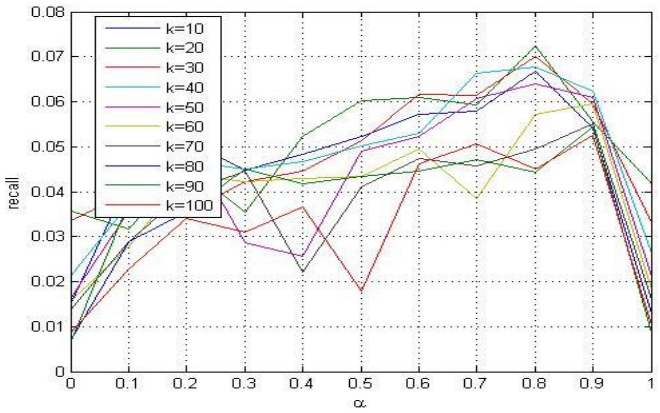

Figure 1. The Effect of the Different $\propto$

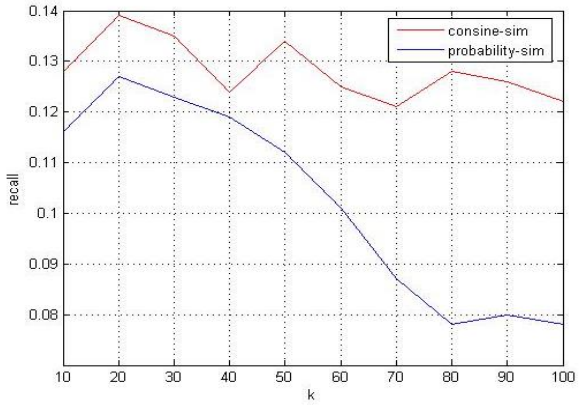

Figure 2.The Effect of the Different Similarities

The Figure 5 shows the comparison of the recall between the group 1 and the group 2 when we use the vector cosine-based similarity to set up the similarity model. We can see from the figure that the group 1 is also better than the group 2, just like the Figure 4 shows. Then we select the k equal to 30 and the result of the comparison shows in the Figure 6.The conclusion is that the conditional probability similarity will get a better result if we choose the group 1. And the results will not be different if we choose the group 2. The Figure 7 shows that the comparison when we use the traditional similarity model and divide the users into two groups. We can see that the group 1 will get the better result no matter which similarity model we use than the result we don't divide the users. The group 2 doesn't show the better result.

And we can see from the Figure 8: With the increasing of k, the recall of two groups is a declining trend. The recall of the two groups is better when the k varies from 10 to 30.The Figure 9 shows the recall of the two groups when we make the use of the conditional probability to set up the similarity model. According to the results, it will not be useful when we divide the users into two groups.

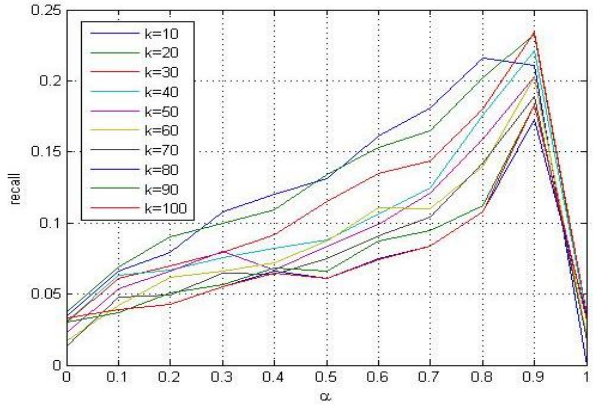

Figure 3.The Effect of the Different $\propto$ to the Group 1

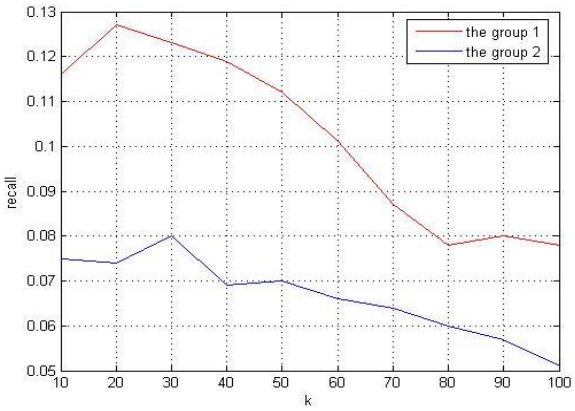

Figure 4.The Comparison of Two Groups based on the Probability Similarity

5. Conclusion

In this study, we analyze the performance of the different recommendation algorithms and propose that the item-based collaborative filtering recommendation algorithms should be applied into the online learning system of the Tianjin University. Our results confirm that when we choose the vector cosine-based similarity and k equals to 30, the result will get the best result. If we choose the conditional probability to set up the similarity model, we will choose the parameter $\propto$ equal to 0.9 and the k equal to 30.

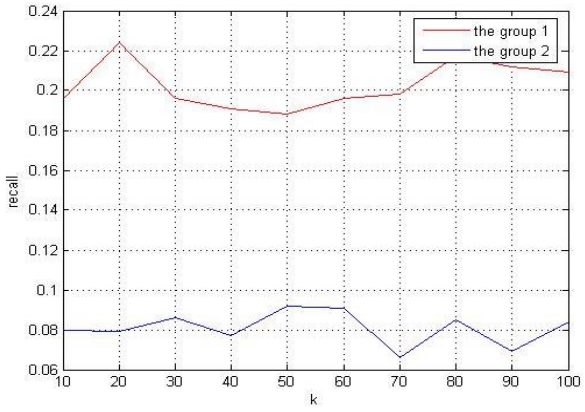

Figure 5. The Comparison of Two Groups based on Cosine Similarity

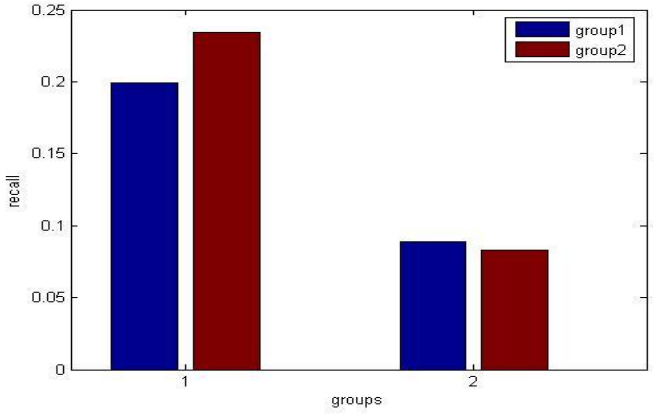

Figure 6. The Effect of the Recall in the Groups based on Different Similarity

Acknowledgements

We would like to thank the anonymous reviewers for their helpful and constructive comments. This work was partly funded by Project 61202030, 61170305 supported by NSFC and Project x2013-005 supported by ECTC.

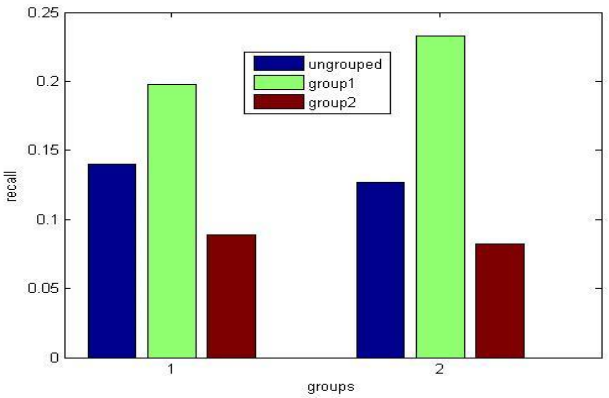

Figure 7. The Comparison of the Traditional Item-based Algorithm

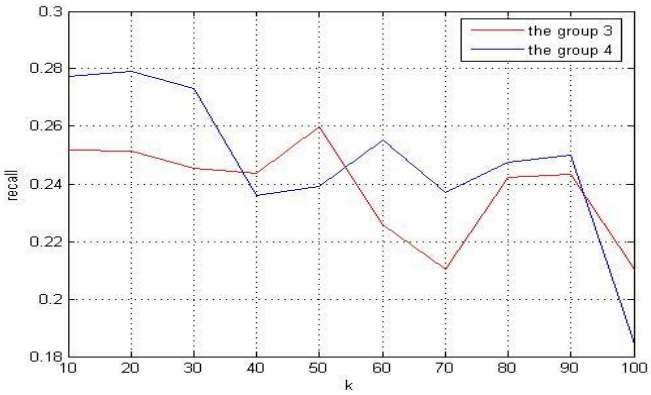

Figure 8.The Effect of the k (Based on the Cosine Similarity)

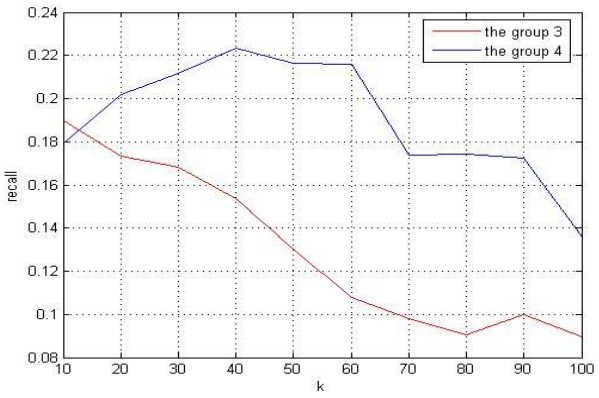

Figure 9. The Effect of the k (based on the Probability Similarity)

References

[1] [1] Wen Y, Shui-shng Y. "A survey of collaborative filtering algorithm applied in E-commerce recommender system ". Computer Technology and Development. (2006).
[2] [2] Resnick P, Iacovou N, Suchak M, et al. "GroupLens: an open architecture for collaborative filtering of netnews". In Proceedings of the 1994 ACM conference on Computer supported cooperative work. (1994).pp. 175–186
[3] [3] Twigg C A. "Models for online learning". Educause Review.(2003).pp.28-38.
[4] [4] Breese J S, Heckerman D, Kadie C."Empirical analysis of predictive algorithms for collaborative filtering". In Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence. (1998). pp.43–52.
[5] [5] Goldberg D, Nichols D, Oki B M, et al. "Using collaborative filtering to weave an information tapestry ". Communications of the ACM. (1992).pp. 61–70.
[6] [6] Hong Wei Ma,Guang Wei Zhang,Peng Li."Survey of Collaborative Filtering Algorithms ". Journal of Chinese Computer Systems. (2009) .pp. 1282–1288.
[7] [7] Sarwar B, Karypis G, Konstan J, et al. "Item-based collaborative filtering recommendation algorithms". In Proceedings of the 10th international conference on World Wide Web. (2001).pp.285–295.
[8] [8] Linden G, Smith B, York J. Amazon.com recommendations. "Item-to-item collaborative filtering". Internet Computing, IEEE, (2003).pp. 76–80.
[9] [9] Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems. "A survey of the state-of-the-art and possible extensions". Knowledge and Data Engineering, IEEE Transactions on. (2005) .pp. 734–749.
[10] [10] Deshpande M, Karypis G. "Item-based top-n recommendation algorithms". ACM Transactions on Information Systems (TOIS). (2004). pp.143–177.
[11] [11] P. Melville, R. J. Mooney, and R. Nagarajan." Content-boosted collaborative filtering for improved recommendations". in Proceedings of the 18th National Conference on Artificial Intelligence (AAAI '02), (2002).pp. 187–192.
[12] [12]Su, Xiaoyuan, and Taghi M. Khoshgoftaar. "A survey of collaborative filtering techniques." Advances in artificial intelligence (2009).

Authors

Ruiguo Yu, he is an associate professor at the School of Computer Science and Technology of Tianjin University. His research interests include machine learning, artificial intelligence and data mining.

Zhiyong Cai, he is a graduate student at the School of Computer Science and Technology of Tianjin University. His research interests include machine learning, information retrieval and data mining.

Xiuping Du, he is an associate professor at the School of Education of Tianjin University. His research interests include educational technology and data mining.

