A Personalization Recommendation Algorithm for E-Commerce

Hui Li
Department of Computer Engineering, Huai Hai institute of Technology, Lianyungang, China
Email: shufanzs@126.com

Shu Zhang
School of Business, Huai Hai institute of Technology, Lianyungang, China
Email: shufanzs@sina.com

Xia Wang
Department of Computer Engineering, Huai Hai institute of Technology, Lianyungang, China
Email: lijunmagic@yahoo.com

***Abstract*—Many recommendation systems employ the collaborative filtering technology, which has been proved to be one of the most successful techniques in recommendation systems in recent years, the difficulties of the extreme sparsity of user rating data have become more and more severe. To solve the problems of scalability and sparsity in the collaborative filtering, this paper proposed a personalization recommendation algorithm based on rough set which is proposed, The algorithm refine the user ratings data with dimensionality reduction, then uses a new similarity measure to find the target users' neighbors, then generates recommendations. To prove our algorithm's effectiveness, the authors conduct experiments on the public dataset. Theoretical analysis and experimental results show that this method is efficient and effective**

***Index Terms*—e-commerce, recommendation, deduction, algorithms**

I. INTRODUCTION

With rapidly increasing amount of information in the networks, there is a serious need for a new technology to help people find what they want from a huge mass of data. Personality service system emerges as the times requires, which is used to help users find the information they are interested in. The provided personalized service is accepted by more and more E-commerce Web site, digital library and many other similar fields, it also becomes one of the most important functions in these systems. At present, almost all large-scale e-commerce systems, such as Amazon, eBay, and taobao, use recommendation systems in a variety of modes.

As a type of information technology that aim to support personalized service, recommendation systems are widely used by e-commerce practitioners and have become an important research topic in information sciences and decision support systems . Recommendation systems are decision aids that analyze customer's prior online behavior and present information on products to match customer's preferences. Through analyzing the patron's purchase history or communicating with them, recommendation systems employ quantitative and qualitative methods to discover the products that best suit the customer. Most of the current recommendation systems recommend products that have a high probability of being purchased [1]. They employ content-based filtering (CBF) [2], collaborative filtering (CF) [3-4], and other data mining techniques, for example, decision tree [5], association rule [7], and semantic approach [7].

Many researchers have proposed various kinds of CF technologies to make a quality recommendation [8-9]. All of them make a recommendation based on the same data structure as user-item matrix having users and items consisting of their rating scores. There are two methods in CF as user based collaborative filtering and item based collaborative filtering [10]. User based CF assumes that a good way to find a certain user's interesting item is to find other users who have a similar interest. So, at first, it tries to find the user's neighbors based on user similarities and then combine the neighbor users' rating scores, which have previously been expressed, by similarity weighted averaging. And item based CF fundamentally has the same scheme with user based CF. It looks into a set of items; the target user has already rated and computes how similar they are to the target item under recommendation. After that, it also combines his previous preferences based on these item similarities.

The traditional collaborative filtering algorithm works by building a database of preferences for items by users. To find information that the target user may probably be interested in, we first discover the target user's nearest neighbors, which are other users who have historically had similar taste to the target. The traditional nearest-neighbor collaborative filtering recommendation algorithms face the challenge of extreme sparsity of user rating data.

To solve the difficulties of the extreme sparsity of user rating data, in this paper, we first refine the user ratings data with dimensionality reduction aiming at solving the problems of sparsity in the collaborative filtering, then uses a new similarity measure to find the target users'

doi:10.4304/jsw.8.1.176-183

neighbors. The experimental results show that the performance of the present item-based collaborative filtering algorithm is improved, even with extreme sparsity of data.

II. PRELIMINARIES KNOWLEDGE

The task of the traditional collaborative filtering recommendation algorithm concerns the prediction of the target user's rating for the target item based on the users' ratings on observed items when the user has not given the rating. Each user is represented by item-rating pairs, and we use $M * N$ matrix to denote the user-item table, which contains the ratings $R(M, N)$ that have been provided by the m*th* user for the n*th* item, the table as following.

TABLE I.

USER-ITEM RATINGS TABLE

	Ite	…	Ite	…	Ite
Use	$R_{1,1}$	…	$R_{1,j}$	…	$R_{1,N}$
…	…	…	…	…	…
Use	$R_{i,1}$		$R_{i,j}$		$R_{i,N}$
…	…	…	…	…	…
Use	$R_{M,}$	…	$R_{M,j}$	…	$R_{M,}$

Where R_{ij} denotes the score of item *j* rated by an active user *i*. If user *i* has not rated item *j*, then R_{ij} =0. The symbol m denotes the total number of users, and n denotes the total number of items.

The recommendation process based on collaborative filtering can be divided into two steps: search the nearest neighborhood and produce recommendation collection.

Step 1: search the nearest neighborhood collection

The core of collaborative filtering algorithm is to find the nearest neighborhood of the target user based on the user's rating matrix. That is, given a present user u, the aim is to produce a neighborhood collection $N=\{N_1,N_2,\ldots N_K\}$ ordering by the similarity among users from big to small, that is $Sim(u, N_i) > Sim(u, N_{i+1})$, $i = 1,..., k-1$.

There are several similarity algorithms that have been used in the CF recommendation algorithm [8-10]: Pearson correlation, cosine vector similarity, adjusted cosine vector similarity, and Spearman correlation. The cosine measure is presented as following formula.

$$Sim(i, j) = \frac{\sum_{k=1}^{n} R_{ik} R_{jk}}{\sqrt{\sum_{k=1}^{n} R_{ik}{}^2 \sum_{k=1}^{n} R_{jk}{}^2}} \quad (1)$$

Where R_{ik} is the rating of the item k by user *i* and n is the number of items co-rated by both users.

Pearson's correlation, as following formula, measures the linear correlation between two vectors of ratings.

$$Sim(i, j) = \frac{\sum_{k \in I_{i,j}} (R_{i,k} - \bar{R}_i)(R_{j,k} - \bar{R}_j)}{\sqrt{\sum_{k \in I_{i,j}} (R_{i,k} - \bar{R}_i)^2} \sqrt{\sum_{k \in I_{i,j}} (R_{j,k} - \bar{R}_j)^2}} \quad (2)$$

Where $R_{i,k}$ is the rating of the item k by user *i*, $\bar{R}_i$ is the average rating of user *i* for all the co-rated. I_{ij} is the items set both rating by user *i* and user *j*.

Step 2: produce the recommendation collection

Since we have got the neighborhood of user, we can calculate the weighted average of neighbors' ratings, weighted by their similarity to the target user. The rating of the target user *i* to the target item k is as following:

$$P_{i,k} = \bar{R}_i + \frac{\sum_{m=1}^{n} sim(i,m) * (R_{m,k} - \bar{R}_m)}{\sum_{m=1}^{n} sim(i,m)} \quad (3)$$

Where $\bar{R}_i$ is the average rating of the target user *i* to the items, sim (i,m) is the similarity degree of the neighbor user *i* and the neighbor user m, Rm,k is rating of user m to the item k, $\bar{R}_m$ is the average rating of user m, n is the number of neighbors.

III. RELATED WORKS

A recommendation system is basically a system that can learn about a user's personal preferences based on the user's characteristics and behaviors and can then provides the most appropriate content to meet the user's needs. Recommendation systems have been applied in various industries, and their usefulness has been recognized in recent years. There are various ways of designing recommendation systems. Different machine learning algorithms can be used to construct a mapping from the features of an item to a number indicating the utility of the item to the user, based on previous ratings that the user has made on other items.

Most recommendation systems use two techniques: content-based filtering and collaborative filtering. Content-based filtering is based on analysis of the content of the objects under consideration and its relation to the user's preferences. For content-based filtering, it is therefore necessary that the results of content analysis and user preferences be reliably determined. One of the most successful technologies used for recommendation systems is collaborative filtering. The basic premise is that users with similar tastes tend to like similar types of items and that consequently a rating by someone similar is a good predictor for a user's rating of an item. Collaborative filtering can improve a recommendation system by taking advantage of this information

Xue, G. et al.[11] present a novel approach that combines the advantages of memory based collaborative filtering and model based collaborative filtering of approaches by introducing a smoothing-based method. In their approach, clusters generated from the training data

provide the basis for data smoothing and neighborhood selection. As a result, they provide higher accuracy as well as increased efficiency in recommendations. Their empirical studies on two datasets as EachMovie and MovieLens show that their new proposed approach consistently outperforms other user based traditional collaborative filtering algorithms.

George, T. et al. [12] consider a novel collaborative filtering approach based on a recently proposed weighted co-clustering algorithm that involves simultaneous clustering of users and items. They design incremental and parallel versions of the co-clustering algorithm and use it to build an efficient real-time collaborative filtering framework. Their empirical evaluation of the proposed approach on large movie and book rating datasets demonstrates that it is possible to obtain accuracy comparable to that of the correlation and matrix factorization based approaches at a much lower computational cost.

Rashid, A.M. et al. [13] propose ClustKnn, a simple and intuitive algorithm that is well suited for large data sets. The proposed method first compresses data tremendously by building a straightforward but efficient clustering model. Recommendations are then generated quickly by using a simple Nearest Neighbor-based approach. They demonstrate the feasibility of ClustKnn both analytically and empirically. They also show, by comparing with a number of other popular collaborative filtering algorithms that, apart from being highly scalable and intuitive, ClustKnn provides very good recommender accuracy as well.

Cantador, I. et al. [14] propose a multilayered semantic social network model that offers different views of common interests underlying a community of people. The applicability of the proposed model to a collaborative filtering system is empirically studied. Starting from a number of ontology-based user profiles and taking into account their common preferences, they automatically cluster the domain concept space. With the obtained semantic clusters, similarities among individuals are identified at multiple semantic preference layers, and emergent, layered social networks are defined, suitable to be used in collaborative environments and content recommenders.

Panagiotis Symeonidis et al. [15, 16] use bi-clustering to disclose this duality between users and items, by grouping them in both dimensions simultaneously. They propose a novel nearest bi-clusters collaborative filtering algorithm, which uses a new similarity measure that achieves partial matching of users' preferences. They apply nearest bi-clusters in combination with two different types of bi-clustering algorithms Bimax and xMotif for constant and coherent biclustering, respectively. Extensive performance evaluation results in three real-life data sets are provided, which show that the proposed method improves substantially the performance of the CF process.

IV. ATTRIBUTION DEDUCTION

The task of the traditional collaborative filtering recommendation algorithm concerns the prediction of the target user's rating for the target item based on the users' ratings on observed items when the user has not given the rating. We can deem the user' rating table as a $M * N$ matrix, which M denote the number of user, N denote the number of ratings item.

The $M * N$ matrix can be deemed as a incomplete information system or a decision table. It is well known that the rating matrix of the user is too sparse to compute the neighbor set of the target user effectively. To solve the problem of rating matrix sparsity, it is necessary to purify the rating matrix. The attribute reduction is a effective method.

In this paper, we present reduction algorithms based on the principle of Skowron's discernibility matrix [17]. The information in the information table (also called decision table) relevant to attribute discriminate are concentrated in a matrix (called Discernibility Matrix) in such method, and to calculate the core attribute through the discernibility matrix. Our core attribute selection algorithm is based on the discernibility matrix, we also study the other attribute combination as well as the core attribute in the discernibility matrix, and utilize disjunctive normal form to conduct attribute deduction.

Skowron[1991] propose the method of using discernibility matrix to express knowledge. This expression has many advantages, especially can be used to interpret and compute data core and deduction. The discernibility matrix is defined as:

Select the simplest attribute combination and add it to a set which severed as the Output attribute set.

Formally, a data set or an information system is a quadruple $S =< U, A, V, f >$,where U is a non-empty finite set of objects, called a universe, A is a non-empty finite set of features, V is the union of feature domains, $f : U \times A \to V$ is an information function ,which make $\forall X \in U, a_i \in A, f(X, a_i) \in V_{a_i}$ .We can split set A of features into two subsets: C $\subset$ A and D = A-C, conditional set of features and decision features ,respectively. Information system can be written briefly IS=(U,A).

Definition 1: Given a information system $IS = (U, A)$, $U = \{x_1, ..., x_n\}$, we can split set A of features into two subsets: conditional set of features $C = \{c_1, ..., c_m\}$ and decision features D: A=C $\cup$ D. Let $c_i(x_j)$ and $D(x_j)$ express the value of data point x_j on conditional attribute set C and decision attribute set D respectively. The value of every element in the discernibility matrix is defined as follows:

$$M_{i,j}=\begin{cases}0, & D(x_i)=D(x_j)\\ -1, & \forall c\in C, c(x_i)=c(x_j), D(x_i)\neq D(x_j)\\ \{c\in C: c(x_i)\neq c(x_j)\}, & D(x_i)\neq D(x_j)\end{cases} \quad (4)$$

$i, j = 1,...,n$

This matrix states that the element value is 0 when the decision attribute is the same; the element value is different attribute combination when the decision attribute is different but can be distinguished by some condition attribute; the element value is negative one when the decision attribute is different but the condition attribute is the same which states that the data is wrong or the condition attribute is insufficient.

From the definition of the discernibility matrix, we can draw a conclusion that when the number of attribution combination equals one, which indicates that the other attributions besides this attribute can not distinguish two records, so such attribute must be preserved. So the attribute with the number of attribution combination equal one in the discernibility matrix constitute the core attribute, which is denoted by C_0 , and $C_0 \subseteq A$.

Applying the attribute reduction to the rating matrix, the sparsity of the rating matrix can be decreased greatly. The experimental result indicates that computation of neighbor set can draw more exactly result on the purified matrix.

V. ATTRIBUTION DEDUCTION

With the development of E-commerce, the user rating data is more and more sparse, and the quantity of items that have been rated by the same users is limited. The user-based CF needs to find at least two users who have rated at least two same items. The fact is similar users may not be able to be found if they lack same rated items. So the traditional similarity measuring method can not work well in the state of sparse database. In this paper, we propose a new method to search the nearest neighbor set of the target user through computing the distance to all the other users in the rating matrix.

In this paper, we introduce the concept of neighbor domain in rough set into the recommendation system. Following we redefine and interpret the concept of information system in the rough set. Given an information system $S =< U, A, V, f >$, $U = \{x_1, x_2,..., x_m\}$ denotes the universe of discourse composed by all the users(not include the target user x_0), $A = \{a_1, a_2,...a_{n-1}\} \cup \{a_n\}$, $\{a_n\}$ denotes the decision attribute, which stand for the not rating item of the target user x_0 , $\{a_1, a_2,...a_{n-1}\}$ denotes the conditional attributes, which stand for the rating item of the target user x_0 ; V denotes the ratings values set, the information function $f : U \times A \to V$ denotes every user's ratings to item, $a_k(x_l)$ denotes user x_l gives the ratings of item a_k .If the target user x_0 can not score the item a_i ,then set $a_i(x_0) = 0$ .So the user set U and item set A compose a user-item rating matrix R(m, n) ,following we give the definition to the near neighbor of user x .

Definition 2: Given a distance function $D : f(x, y) \to R+$, where $R+$ denotes positive integer set, to every $x \in U$, $B \subseteq C$,and $q \in R+$,the near neighbor $n_B^q(x)$ of user x in sub space B is defined as following:

$$n_B^q = \{y \mid x, y \in U, D_B(x, y) \le q\} \quad (5)$$

D is a distance function, generally the Manhattan or the Euclidean distance function is widely used.

The value difference metric (VDM) was introduced by Stanfill and Waltz to compute the nearest neighbor of the target user. A simple version of the VDM is defined as follows:

$$VDM(x, y) = \sum_{a \in A} d_a(x_a, y_a) \quad (6)$$

Where A is the set of rating item , x and y are any two objects between which we shall calculate the distance and $d_a(x_a, y_a)$ denotes the distance between two values x_a and y_a , where x_a is the rating of object x on item a, y_a is the rating of object y on item a .

For any item a in set A, $d_a(x_a, y_a)$ is defined as follows:

$$d_a(x_a, y_a) = p(x_a) - p(y_a) \quad (7)$$

Where $p(x_a)$ is the probability of object x on item a and $p(y_a)$ is the probability of object y on item a .

In order to ensure that the value different metric can describe the nearest neighbor of target, we redefine the formula of the value different metric. Next we give the revised definition of VDM and distance in rough set theory under the neighborhood relation.

Definition 3: Given an information system $IS = (U, A)$, where U is a non-empty finite set of objects, called a universe, A is a non-empty finite set of features. Let $x, y \in U$ be any two objects between which we shall calculate the distance. The value difference metric in rough set theory under the neighborhood relation is defined as follows :

$$VDM_{NB}(x, y) = \sum_{a \in A} d'_a(x_a, y_a) \quad (8)$$

Definition 4: For any $x, y \in U$, $a \in A$, let q_a is a neighborhood parameter, the distance between two objects x and y on item a is defined as follows:

$$d'_a(x_a, y_a) = \frac{\left|n_a^{q_a}(x)\right|}{|U|} - \frac{\left|n_a^{q_a}(y)\right|}{|U|} \quad (9)$$

Where $n_a^{q_a}(x)$ is a neighborhood of object x on item a , $n_a^{q_a}(y)$ is a neighborhood of object y on item a .

Repeat the above calculation, we can finally obtain distances for all the other pairs of objects in U. By then, we define a neighborhood-based factor (NBF), which indicates the degree of neighborhood for every object in an information system, and then we can choose the biggest top N to be the nearest neighbors of target user.

Definition 5: NBF: Given an information system $IS = (U, A)$, where U is a non-empty finite set of objects, called a universe, A is a non-empty finite set of features. For any $x \in U$, the NBF of object x is defined as follows：

$$NBF(x_i) = \sum_{j=1, j \neq i}^{n} VDM_{NB}(x_i, y_j) \tag{10}$$

After getting the nearest neighbors of the target item by using the above method, the next step is prediction computation. We use NBS_u to denote the nearest neighbors set of user u, so the prediction rating P_{ui} of u on item i can be computed by the rating of u on item in the nearest neighbors set NBS_u ,the computation formula is as follows：

$$P_{u,i} = \overline{R}_u + \frac{\sum_{n \in NBS_u} sim(u,n) \times (R_{m,i} - \overline{R}_n)}{\sum_{n \in NBS_u} (|\, sim(u,n)\,|)} \tag{11}$$

Where R_{ni} is the rating of user n on item i, $\overline{R}_u$ and $\overline{R}_n$ denote the average rating of user u and user n on item i. $sim(u,n)$ denotes the similarity between user u and user n and the formula is as follows:

$$Sim(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} * \vec{j}}{|\, \vec{i} \,||\, \vec{j} \,|} \tag{12}$$

Where $\vec{i}, \vec{j}$ denote the rating vector of user i, j.

VI. Experiment Results

In this section, we describe the dataset, metrics and methodology for the comparison between traditional and proposed collaborative filtering algorithm, and present the results of our experiments.

A. Date Set and Evaluation Metrics

The data set we used was collected through the MovieLens Web site(movielens.umn.edu) during the seven-month period from September 19,1997 to April 22, 1998.The data set includes totally 100 000 ratings. There are 943 users and 1682 movies, so the data set was converted into a user-item matrix that had 943 rows and 1682 columns. Each user in this data set had rated 20 movies at least, and the scores is an integer from one to five. We divided the database into training sets and test sets. We repeated our experiment in training set and test set, then averaged the results.

The measures for evaluating the quality of a recommender system can be mainly categorized into two classes: statistical accuracy metrics and decision support accuracy metrics [18]. Mean absolute error (MAE) between ratings and predictions is a widely-used statistical accuracy metric. MAE is easy to understand and can measure the quality of recommender systems in an intuitive way. In this paper, we used MAE as the evaluation metrics.

MAE is a measure of the deviation of recommendations from their true user-specified values. The prediction set of user rating is expressed as $\{p_1, p_2, ..., p_N\}$, the corresponding factual user rating set is expressed as $\{q_1, q_2, ..., q_N\}$, formally,

$$MAE = \frac{\sum_{i \in N} |\, P_i - q_i \,|}{N} \tag{13}$$

Where p_i represents the degree of satisfaction that the customer assesses the product, q_i represents the degree of the satisfaction that recommendation algorithm assesses the product, and N represents the total customers. MAE represents the mean absolute error between the real ratings items and the predicable rating items. The more decreased the MAE is, the more the quality of recommendation is increased.

B. The Experiment Result and Analysis

To test the efficiency of the algorithm proposed in this paper, we need to test from two aspects. One side is to test the efficiency of dimension deduction, and the other side is to test the recommendation quality of algorithm proposed in this paper.

To measure the effect of similarity computing method with different ways of filling matrix of the user ratings data, this paper selects two different ways of filling the data set as the test set, that is, extreme sparse data set (set the item score without rating to 0) and SVD Forecast score data sets (the dimension deduction methods used in this paper). We adopt three traditional similarity algorithms basic cosine (Cos), correlation(Pearson) and adjusted cosine (ACos)[19] to compute the nearest neighbor set and tested them on our data sets by computing MAE.

(1) Extreme sparse data set

To assess the validity of the method of the similarity, the four algorithms Yun [20], Cos, ACos and Pearson can be used separately to calculate the nearest neighbor set and tested them on our data sets by computing MAE. Because the original customer rating matrix is very sparse and exists many vacancies value, the accuracy of the similarity calculation is very low in such data set. The commonly used method is using some fixed value to fill the vacancy value. Table 2 shows the MAE result with different Similarity calculation method when the item score without rating is set to 0. The results show that the method of using direct filling 0 get the higher MAE, the prediction effect of 4 methods are not good: the effect of Pearson is the worst,ACos take the second place, the effect of Cos an Yun is good, Cos is the best. The value of MAE is between 3.0 and 3.6. The experimental result indicates that the value of the filling with fixed effect cannot obtain very good prediction effect, so the SVD

technology is used to simplify the dimension of the score matrix in this paper.

TABLE II.

THE MAE USING DIFFERENT SIMILARITY COMPUTE METHOD IN EXTREME SPARSE DATE SET

N	*cos*	*ACos*	*Pearson*	*Yun*
N=5	3.01	3.37	3.45	3.07
N=10	3.01	3.30	3.57	3.03
N=15	3.02	3.26	3.59	3.05
N=20	3.03	3.25	3.58	3.06
N=25	3.04	3.25	3.56	3.08
N=30	3.04	3.23	3.55	3.09

(2) SVD Forecast score data sets

We need to determine the dimension K of the singular value decomposition before using SVD algorism. We experiment many times in the initial test set, the K form 1 to 25 increasing, step of 1. The results show that, when k = 8, the algorithm is minimum MAE 0.83, the forecast precision is high. Therefore, we keep dimension k of SVD to 8.

The four similarity compute methods mentioned above are used separately to compute the neighbor set after assuring the dimension k. The results indicate that applying the dimension deduction before computing the neighbor set will get higher recommendation efficiency. As table 3 shows, all the MAE value between 0.80 and 0.82, and the minimum MAE value is 0.79 which achieve optimal. Experimental results show that the application of SVD dimension simplified is practicable and effective.

TABLE III.

THE MAE USING DIFFERENT SIMILARITY COMPUTE METHOD IN SVD DATE SET

N	Cos	ACos	Pearson	Yun
N=5	0.840	0.818	0.804	0.819
N=10	0.822	0.818	0.822	0.810
N=15	0.821	0.817	0.821	0.802
N=20	0.818	0.817	0.818	0.800
N=25	0.818	0.816	0.818	0.800
N=30	0.820	0.816	0.820	0.800

We compare the proposed method with the traditional collaborative filtering. In our experiments, we vary the number of neighbors and compute the MAE. We implemented three traditional similarity algorithms basic cosine (Cos), correlation (Pearson) and adjusted cosine (ACos) and tested them on our data sets by computing MAE. The neighbor number is from 5 to 30. In our experiments, we vary the number of neighbors and compute the MAE. The obvious conclusion from Figure 1, which includes the Mean Absolute Errors for the proposed algorithm and the traditional collaborative filtering as observed in relation to the different numbers of neighbors, is that our proposed algorithm is better.

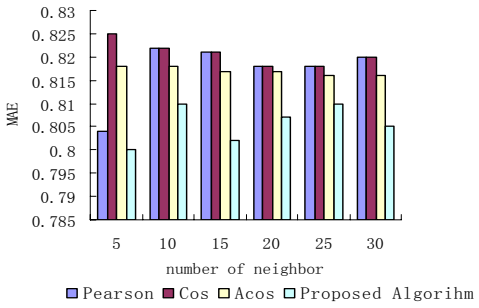

Figure 1. Comparison of Comparing the proposed CF algorithm with the traditional CF algorithm.

In the traditional collaborative filtering recommendation algorithm, if correlation or adjusted cosine is used to measure similarity, the similarity between two items is determined by the ratings of users who have rated both these items. However, with the extremely sparse data, the quantity of these users is very limited, so even these users have similar ratings on these two items, these two items may not be the exact nearest neighbors. The experiment results support that the accuracy of recommendations is poor. Using basic cosine to measure similarity, there is no statistics information of items, and if there were two items that had never been rated by any user, these two items are regarded as similar with this method. This result is obviously imprecise. The proposed algorithm first refines the user ratings data with dimensionality reduction which aims at solving the problems of sparsity in the collaborative filtering, then uses a new similarity measure to find the target users' neighbors. So the proposed algorithm can have better performance.

VII. CONCLUSION

Recommender systems can help people to find interesting things and they are widely used in our life with the development of electronic commerce. Many recommendation systems employ the collaborative filtering technology, which has been proved to be one of the most successful techniques in recommender systems in recent years. With the gradual increase of customers and products in electronic commerce systems, the time consuming nearest neighbor collaborative filtering search of the target customer in the total customer space resulted in the failure of ensuring the real time requirement of recommender system. At the same time, it suffers from its poor quality when the number of the records in the user database increases. Sparsity of source data set is the major reason of causing the poor quality. To solve the problems of scalability and sparsity in the collaborative filtering, this paper proposed a personalized recommendation approach joins the rough set technology and nearest neighbors. This method first reduces the sparsity of rating matrix by attribution deduction and based on the similarity between target user, the nearest neighbors of target user can be found and smooth the

prediction when necessary. Then, the proposed approach is more scalable and more accurate than the traditional one.

Despite there are some contribution of this research, there are limitations, further works can be done. The most important work is to investigate the factors that impact a customer's feelings. Many attributes such as the demographic and psychological characteristics, purchase and consumption environment, and customers' expectation, may well have significant influence on customers' feelings toward a specific product. Therefore, it is crucial to identify the factors important for modeling rating classification, so as to predict the customer's satisfaction level effectively.

Another work is to elicit customers' needs and preferences. The rating classification aims to recommend the right products based on customers' characteristics to achieve high satisfaction levels. Therefore, the validity of customers' need and preferences has an important implication on the effectiveness of the recommendation system. Oftentimes consumers do not have clear need and preferences. Therefore, finding an effective way to facilitate customers to express their true need and preferences is essential for the recommendation systems.

ACKNOWLEDGMENT

The research is supported by the Natural Science Foundation of Jiangsu Province Nos. BK2008190, and technology open fund project of Jiangsu Province R&D Institute of Marine Resources. No. JSIMR11B12.

REFERENCES

[1] A.V.Bodaptati, “Recommendation systems with purchase data, Journal of Marketing Research”, 45 (1), pp.77-93, 2008.

[2] A.Zenebe, A.F.Norcio, “Representation, similarity measures and aggregation methods using fuzzy sets for content-based recommender systems”, Fuzzy Sets and Systems, 160 (1), pp. 76-94, 2009.

[3] J.L. Herlocker, J.A. Konstan, J. Loren, G. Terveen, T. Riedl, “Collaborative filtering recommender systems”, ACM Transactions on Information Systems, 22 (1), pp. 5-53, 2004.

[4] H.W. Ye, “A Personalized collaborative filtering recommendation using association rules mining and self-organizing map”, Journal of Software, 6(4), pp.732-739, 2011.

[5] Y.M. Zhang, S.Y. Jiang, “A Spliting criteria based on similarity in decision tree learning”, Journal of Software, 7(8), pp.1775-1782, 2012.

[6] F.H. Wang, H.M. Shao, “Effective personalized recommendation based on timeframed navigation clustering and association mining”, Expert Systems with Applications, 27 (3), pp.365-377, 2004.

[7] T.P. Liang, Y.F. Yang, D.N. Chen, Y.C. Ku, “A semantic-expansion approach to personalized knowledge recommendation”, Decision Support Systems, 45 (3), pp. 401-412, 2008.

[8] C. Li, “A Hybrid Item-based Recommendation Algorithm against Segment Attack in Collaborative Filtering Systems”. In: Proceedings of the Information Management, Innovation Management and Industrial Engineering (ICIII2011), pp. 403-406, 2011.

[9] C.B Huang, S.J. Gong, Employing rough set theory to alleviate the sparsity issue in recommender system, In: Proceeding of the Seventh International Conference on Machine Learning and Cybernetics (ICMLC2008), IEEE Press, pp.1610-1614, 2008.

[10] L. Ren, J.Z Gu, W.W. Xia, “An Item-Based Collaborative Filtering Algorithm Utilizing the Average Rating for Items”, Signal processing and multimedia, vol.123, pp.175-183, 2010.

[11] G. Xue, C. Lin, Q. Yang, et al. “Scalable collaborative filtering using cluster-based smoothing”, In Proceedings of the ACM SIGIR Conference, pp.114-121,2005.

[12] T. George, S. Merugu, “A scalable collaborative filtering framework based on co-clustering”. In Proceedings of the IEEE ICDM Conference, pp. 27-30, 2005.

[13] A.M. Rashid, S.K. Lam, G. Karypis, J. Riedl, “ClustKNN: A Highly Scalable Hybrid Model- & Memory-Based CF Algorithm”, In proceedings of the IEEE WEBKDD 2006.

[14] I. Cantador, P. Castells, “Multilayered Semantic Social Networks Modeling by Ontology-based User Profiles Clustering: Application to Collaborative Filtering”. Managing Knowledge in a World of Networks , vol.4248, pp.334-349, 2006.

[15] P. Symeonidis, A. Nanopoulos, A. Papadopoulos, Y. Manolopoulos, “Nearest- Biclusters Collaborative Filtering”, In Proceeding of the IEEE WEBKDD 2006.

[16] P. Symeonidis, A. Nanopoulos, A. Papadopoulos, Y. Manolopoulos, “Nearest-biclusters collaborative filtering based on constant and coherent values”, Information retrieval, vol.11,no.1, pp.51-75, 2007.

[17] A. Skowron, C. Rauszer, “The discernibility matrices and functions in information systems”, Handbook of application and advances of rough set theory, vol.11, pp. 331-362, 1992.

[18] M. Deshpande, G. Karypis, “Item-Based Top-N Recommendation Algorithms”, ACM Trans Information Systems, vol.22, no.1, pp.143-177, 2004.

[19] B. Sarwar, G. Karypis, J. Konstan, “item-based collaborative filtering recommendation algorithms”, In Proceedng of the 10th international conference on World Wide Web, pp. 285-295, 2001.

[20] Y. Koren, “Factor in the Neighbors: Scalable and accurate collaborative filtering”, In Proceeding of ACM Transactions on Knowledge Discovery from Data, 4(1), pp.1-24, 2009.

Hui Li was born in Lianyungang, JiangSu Province, P.R.China, in October 20, 1979. She received her BE degree in computer science and technology from the Yang Zhou University, in 2007. She is a master candidate in the School of Information & Electrical Engineering in China University of Mining & Technology. She is currently a teacher in Department of Computer Engineering, Huai Hai institute of Technology, Lianyungang, JiangSu Province, P.R.China.

Her current research interest includes data mining, information processing and intelligent computing. He has

published more than 20 papers in journals and conferences.

Shu Zhang, born in 1979. Received her BE degree in computer science and technology from the Yang Zhou University, in 2007. He is currently a teacher in school of business, Huai Hai institute of Technology, Lianyungang, JiangSu Province, P.R.China. His current research interests include data mining and e-commerce.

Xia Wang, born in 1977. Received her BE degree in computer science and technology from the Yang Zhou University, in 2011. She is currently a teacher in Department of Computer Engineering, Huai Hai institute of Technology, Lianyungang, JiangSu Province, P.R.China. Her current research interests include data mining and information retrieval.