INT J COMPUT COMMUN, ISSN 1841-9836
8(4):560-570, August, 2013.

A Model for Collaborative Filtering Recommendation in E-Commerce Environment

Y. Jing, H. Liu

Yuanchun Jing & **Honglu Liu**
School of Traffic and Transportation
Beijing Jiaotong University
PO Box 100044, Beijing, China
12121021@bjtu.edu.cn, hlliu@bjtu.edu.cn

Abstract: In modern business environment, product life cycle gets shorter and the customer's buying preference changes over time. Time plays a more and more important role in collaborative filtering. However, there is a gap in one class collaborative filtering (OCCF). On the basis of collecting different real-time information, this paper proposes an optimization model for e-retailers. Through comparing different methods with different weights, results show that real-time dependent in OCCF performs better in improving the quality of recommendation. The model is effective in cross-selling e-commerce, personalized, targeted recommendation sales.
Keywords: Integration of real-time information, one class collaborative filtering, e-commerce.

1 Introduction

With the rapid development of Internet and the increase of the amount of data on the Internet, Retail businesses face many challenges, such as assessment of customer interests to products, as well as an accurate assessment of products at the market's perspective in modern life. These challenges in the transaction process, often translate into how to find the best match between the customer and the product. In recent years, many companies began to get involved in e-commerce mode, making great progress. According to the "2012 China online retail market data monitoring report" released by the third-party e-commerce research institutes, China Electronic Commerce Research Center (100EC.cn): China online retail market transactions reached 1.3205 trillion RMB, attaining a year-on-year growth of 64.7% by December, 2012; this report also predicts: online retail market turnover of 2013 is expected to reach 1.8155 trillion RMB. As of 2012, China's online retail market transactions have been accounted for 6.3% of the same period's total retail sales of social consumer goods, while the proportion data was only 4.4% in 2011, which means that e-commerce has begun to change the retail landscape.

Despite a huge opportunity of e-commerce development, retailers have to solve complex problems which at a higher level. i.e., in recent years, because of the increase of customers and product, the retailers need to develop a systematic recommendation list. The list may be a list of items for the target customer, or a customer list for the target item. Collaborative filtering (CF) is the method that an associated list generated by the information from customer clearly expressed their interests or extracted from the potential customer behavior information. One of the most common research problems is the Netflix movie recommendation in this area.

Through huge amount of datasets, like purchases transaction datasets, news recommendation records of recommended sites, the problem to be solved can be converted to one class collaborative filtering (OCCF) problems. One class collaborative filtering is a problem with fewer researches, the effectiveness based on one class collaborative filtering depends on the consideration and processing of non-labeled or missing dataset. i.e., its basic idea is to construct a negative dataset (negative sample). As for the one class collaborative filtering problems which have weight distribution application for the matrix factorization technique, recent studies are often based on frequency information, such as customer number, product number, product popularity.

In spite of extensive literature researches we found little study in which similar real-time information is combined, undeniable, real-time plays an important role in the OCCF. Therefore, in this paper, we built a recommended model which can make full use of the real-time information about product and customer in the market. Through the integration of real-time information including: the information of product launched into the market which is related to product life cycle, the information about product itself, and customer recently visiting information related to the customer relationship management, this model can be divided into two aspects: Real-time customer-oriented weight and Real-time product-oriented weight, to improve the quality of one class collaborative filtering (OCCF) for electronic retailers and meet their demand of the online recommendation.

The rest of this paper is organized as follows: Section 2 contents relevant concepts; in Section 3, we proposed a real-time dependent model, added the weight to the model and compared seven different methods with different weight; Section 4, we use the MovieLens dataset to test the model and get results; in Section 5, conclusion and the future work.

2 Relevant concepts

Nowadays, customers are presented with a majority of options for products and services. On the other hand, a vendor in the e-retailer can reach lots of customers what means that there is no limiting factor between the e-retailers and global customers. So, in today's e-retails' competition, the winner may be the one, which can offer a highly level of personalization to an individual customer. For the personalization needs recommendation from the e-retailers, a good recommendation system is the key factor for them to solve the problem. The recommendation algorithm most widely used in the recommendation system is collaborative filtering, namely, CF.

2.1 Collaborative Filtering (CF)

The kernel of collaborative filtering problem is analysis of user interests, and finding some similar (interest) users for the specified user in the user group. After combining these similar users' evaluations to products and developing a recommendation system with this information, the e-commerce site can make prediction to the specified user preference.

The earliest defined "collaborative filtering" are Goldberg and his partners [1]. They developed the first recommendation system, Tapestry, mainly for coordination and filtering newsgroups streaming media files. Another system to study news item filtration, GroupLens, developed by Resnick, etc., similar to Tapestry. GroupLens is mainly for scoring, to test the level of users' interests on the news item. Based on a similar approach, GroupLens project team then got down to MovieLens project, from the filtration problem of news items transferred to movie recommendation problem.

In recent years, collaborative filtering algorithm has been widely studied at home and abroad. Such as, Li G. and Li L. (2012) [2], Shani G., D. Heckerman, and R. I. Brafman (2005) [3], Banati H. and Mehta S. (2010) [4]. On the basis of different technologies, CF system is mainly divided into two categories, one is based on storage, and the other is model. Pearson correlation approximation method can be applied to the collaborative filtering based on storage technology, this kind of CF can also make use of the item-to-item's Top-N recommendation method, which is widely used in e-commerce, like www.Amazon.com, www.jd.com, etc. that recommended for users of all kinds of goods they may like, such as books, audio-visual products, electrical appliances, clothing.

Model-based CF approach makes up for many deficiencies of the one based on the storage. This method usually takes steps: firstly, find the clear or potentially preferences complex patterns between users and items. Then, collect historical data, and provide recommendation model suggests, such as Bayesian model, relies on the network based on the model, clustering model, and the model based on the MDP.

2.2 One Class Collaborative Filtering (OCCF)

According to different data, collaborative Filtering processes can be divided into two categories: one category is the clear preference data processing, for example the score; the other is implicit data's processing, for instance page hits or not. The later always exists in the real world application environments widely, such as whether the user bought a product, whether the user clicked a web page or not. Because of no requirement that provide a clear score, the users, obtain this data much easier than the ratings data. Besides that, in the implicit datasets, only active factors can be clearly distinguished, while the negative cases are uncertain. So this kind of problem is called one class collaborative filtering (OCCF) problem. One class collaborative filtering's task is to rank recommendation sets according to the user's preference levels by analyzing the implicit information with the particular user's favorite. Despite easy acquisition for such data, it is difficult to explain. For example, as for the data of a user clicks the website, among all data, the data which consists of the user clicks the pages can be interpreted as active factors; the rest of the data is the mix of negative data and missing data. How to explain such combined data, and how to process this data after explanation, are the main difficulties of one class collaborative filtering's current studies.

Compare to the CF, the researches about one class collaborative filtering are fewer. We learned something from those few literatures, for example, Yang S. and Xue W. [4]. As for one class collaborative filtering problems, due to the active dataset rarely (sparsity), while the other two types of datasets, negative and missing datasets, are very confusing, it has many difficulties to future research.

Negative and unobserved data processing is the key issue of one class collaborative filtering. For this issue, a large quantity of research focuses on the technology of weight distribution, because the weighted low rank approximation can improve the quality of unobserved data recommendations. In this aspect, the typical studies contain: Pan, etc. proposed a matrix factorization model based on different weight distribution of potential factors [5]. Sindhwani used the formula simplifies weight distribution scheme, proposed another optimization variables, providing an unobserved data processing measure [6].

2.3 Real-time research

Concerning CF problems, most of models regard users' behaviors as static, for example, a customer who buys item A will buy item B. However, product has its product life cycle, and the customer's preferences will change over time, therefore, to regard these problems as static has certain deficiency. In 2009, Koren proposed a real-time changing model which is based on the evaluation of Netflix recommendation in the report of improving the performance [7]. Recent some related studies began to consider the real-time information, such as, the research of Lu, Agarwal, and Dhillon (2009) [8]; Xiang L. and Yang Q. (2009) [9]; Xiong L., X. Chen, T. Huang, J. Schneider, J. G. Carbonell (2010) [10]. But their studies focus on the CF, few mentioned the OCCF. Even few reports establish the problem that real-time information applied to the system of OCCF.

In one class collaborative filtering field, previous studies have only considered frequency-based information, there is few research which designs real-time information to expand collaborative filtering problem. So, this paper, on the basis of exploration the information of product launched into market and the customers' recent access information, considered the product life cycle (PLC) and Customer Relationship Management (CRM) real-time information Our model shows that predictions accuracy can be improved by OCCF recommendation model of e-commerce based on real-time information. Real-time information integration has significant significance for business processes such as cross-selling, up-selling and others' accurate recommendations.

3 Real-time dependent model

3.1 The notation and formula

To facilitate the study and discuss, firstly, we presume some items as follows.

X: Binary representation of actual transaction matrix, 1 means purchase; 0 is no purchase

U: User feature matrix with latent features of customers

P: Product feature matrix with latent features of products

Y: Recommendation resultant matrix

W: Weight matrix: $\in$ [0, 1]

We are given that there are m customers, n products, then, X matrix has m rows and n columns. If i^{-th} row and j^{-th} column in the X matrix has a value of 1, it means that customer i purchased a product j. If it is 0, it means that this particular customer-product has no purchase record. As for the actual transaction, X matrix is very large, and there is a certain degree of sparsity, X manifests itself in a display that less 1 more 0.

In the matrix, r is the rank; it needs to analyze the number of potential characteristics. Presume U is on behalf of user feature matrix, so, $U = [u_1, u_2, \ldots, u_m]^T$ is a $m \times r$ matrix. i^{-th} in the matrix U means a customer, u_i means customer i in r-dimensional customer feature space. In the same way, supposes that P is in the name of product feature matrix, thus, $P = [p_1, p_2, \ldots, p_n]$ is a $r \times n$ matrix. j^{-th} in the matrix P stands for a product, p_j means product j in r-dimensional product feature space. Typically, there is $m, n \gg r$ and $m \times n \gg (m+n) \times r$. Joint matrix is expressed by $u_i^T p_j$ which represent the joint relationship between customers and products. Let's assume $Y = UP$, therefore, the problem can be converted into the following optimization problem:

$$\mathop{\arg\min}_{U \geqslant 0, P \geqslant 0} \sum L(X, Y) \tag{1}$$

In the formula, L is the square error function or other loss functions:

Square error:

$$L(X, Y) = \|X - Y\|^2 = \sum_{i=1, j=1}^{m,n} (X_{i,j} - Y_{i,j})^2 \tag{2}$$

Relative entropy loss:

$$L(X, Y) = D(X \| Y) = \sum_{i=1, j=1}^{m,n} \left(X_{i,j} \log \frac{X_{i,j}}{Y_{i,j}} - X_{i,j} + Y_{i,j} \right) \tag{3}$$

In order to prevent over-fitting, parameter λ is introduced in the formula (1), to revise the optimization model:

$$\mathop{\arg\min}_{U \geqslant 0, P \geqslant 0} \lambda \left(\|U\|_F^2 + \|P\|_F^2 \right) + \sum L(X, Y) \tag{4}$$

Among it, $\|U\|_F^2$ and $\|P\|_F^2$ are the U and P matrices' F norm.

Collaborative filtering system is primarily intended to provide customers with similar product recommendations. The above-mentioned optimal formula major considers the situation that just value of 1. Thus, In terms of the one class collaborative filtering, this formula is not perfect. In one class collaborative filtering, describing the potential customers' preferences to a variety of products or different importance degrees of different data sets through simply the weights setting.

Therefore, adds the weight to original optimization model:

$$\underset{U\geqslant 0,P\geqslant 0}{\arg\min}\,\lambda\left(\|U\|_F^2+\|P\|_F^2\right)+\sum WL\left(X,Y\right) \tag{5}$$

Srebro and Jakkola applied the low-rank approximation based on weights in collaborative filtering model with two extremes weights [11]: active factors has a weight of 1, other factors have weight of 0. Supposes X^1, $X^1=\{(i,j):X_{i,j}=1\}$ which contains (i,j) pairs corresponding all 1of a matrix X. X^0, $X^0=\{(i,j):X_{i,j}=0\}$ containing (i,j) pairs corresponding all 0 (no purchase) in X matrix.

According to the above weight setting scheme, the formula can be further amended to this:

$$\underset{U\geqslant 0,P\geqslant 0}{\arg\min}\,\lambda\left(\|U\|_F^2+\|P\|_F^2\right)+\sum_{i,j\in X^1}W_{i,j}L\left(X_{i,j},u_i^Tp_j\right) \tag{6}$$

Among it,

$$W_{i,j}=\begin{bmatrix}1 & \forall\,(i,j)\in X^1\\ 0 & \forall\,(i,j)\in X^0\end{bmatrix}$$

However, the model ignores the non-buyers groups. In order to comply with the requirement of one class collaborative filtering that settings different weight for different types of factors, the above formula (6) further revised as follows:

$$\underset{U\geqslant 0,P\geqslant 0}{\arg\min}\,\lambda\left(\|U\|_F^2+\|P\|_F^2\right)+\sum_{i,j\in X^0}W_{i,j}L\left(0,u_i^Tp_j\right)+\sum_{i,j\in X^1}W_{i,j}L\left(1,u_i^Tp_j\right) \tag{7}$$

The formula covering all types of customers by setting the weighting values $W_{i,j}$: good customers, potential customers and non-customers. After setting the corresponding weights, the equation (7) can be rewritten as the following equation, similar to the model provided by Sindhwani:

$$\underset{U\geqslant 0,P\geqslant 0}{\arg\min}\,\lambda\left(\|U\|_F^2+\|P\|_F^2\right)+\|\Omega\otimes(X-UP)\|_F^2 \tag{8}$$

$\Omega_{i,j}=\sqrt{W_{i,j}}$, $\otimes$ means for computing according to element, it is to make the two different corresponding elements within matrices multiplied.

Finally, according to the steps described by Lee and Seung [12], final optimization scheme is made by multiplication replaced:

$$P=P\otimes\frac{U^T\left(\Omega\otimes X\right)}{U^T\left(\Omega\otimes(UP)\right)+\lambda P} \tag{9}$$

$$U=U\otimes\frac{\left(\Omega\otimes X\right)P^T}{\left(\Omega\otimes(UP)\right)P^T+\lambda U} \tag{10}$$

3.2 Weight setting scheme

Time weight applied to the collaborative filtering is more and more popular, we can find it from the previous studies. Such as, the research of Lan W. and Zhengjun Z. (2007) [13]; HuaizhenY., Xiaoqi C. and Meilian. L. (2009) [14]; Donghui L., Dewei P. and Hui, Zh. (2012) [15].

In our research, we compared the real-time weight with basic methods.

1) Basic methods

The basic method used by initializing values and other relevant information can be summarized as follows:

(1) 0 weight: weight matrix is a determine matrix similar to the transaction matrix, 1 is the weight of all transactions entered, 0 is the weight of all non-transaction entries.

(2) Full weight: whole weight matrix is 1. Error function in the formula not only needs to calculate the maximum error of transaction entry, but also need to calculate the maximum error of non-transactional input.

(3) Uniform weight: unlike the two extremes of weight setting, the method is to set a smaller weight $\delta \in (0,1)$ for all non-transaction input.

(4) The customer oriented weight: the weight of non-transaction setting is proportional to customer transactions. Associated customers weight is calculated as follows:

$$\delta_{i,j} \propto \frac{\sum_j^n X_{i,j}}{\left[\max\left(\sum_j^n X_{i,j}\right)\right]}$$

$\delta \in (0,1)$, If a customer has many products' historical purchase records, namely, a lot has been tagged data, that non-trading or unlabeled data, is likely to be treated as a customer non-purchases situation.

(5) The product oriented weight: similar to the method (4), weight setting in proportion to the number of traded products. Calculated as follows:

$$\delta_{i,j} \propto \left\{1, \frac{\sum_i^m X_{i,j}}{\left[\max\left(\sum_i^m X_{i,j}\right)\right]}\right\}$$

After obtained the results, using subtraction allows $\delta \in (0,1)$. The main principle is that if the transactions number of a product is small, and most of situations that the non-trading or missing data will be treated as the case of non-purchases.

2) Real-time weight

(1) Real-time customer-oriented weight: In addition to transaction data set, it also involves the customer's recent visiting. Record the history recently visiting vector as ζ^{CR}, which stands for that customers' recently visited records.

ζ^{CR}, It is the vector that customer recently visited distance; it can be expressed as follows:

$\Delta^{CR} = \tau_c - \zeta^{CR}$ Among it, τ_c is the current or assessment phase.

$$\delta_i^{CR} \propto \left[1 - \frac{\Delta_i^{CR}}{\max\left(\Delta^{CR}\right)}\right] \text{So}, \delta \in (0,1)$$

In the formula, δ_i^{CR} is customer i recently visited value based on the real time and the weight value.

The weight of each customer in real time is δ, $\delta \in (0,1)$. This means that if a customer has recently visited the e-commerce site, then compared to those who have not recently visited the site of all customers, he is more likely to become frequent customer.

(2) Real-time product-oriented weight: Reference different time information that products launched into market, set up a vector ζ^{PL}.

ζ^{PL} It is the vector which stands for the time information that products launched into market. It can be expressed as:

$\Delta^{PL} = \tau_c - \zeta^{PL}$. In the formula, τ_c is the current or assessment phase.

$$\delta_j^{PL} \propto \left[\frac{\Delta_j^{PL}}{\max\left(\Delta^{PL}\right)}\right] \text{So}, \delta \in (0,1)$$

δ_j^{PL}, it is the weight setting that product j based on the real time launched into the market.

If a product exists long time in the market, it will attract many customers. Because of innovation emerging, imperfect goods will have a longer product life cycle. In any retail establishment, the records that products launched into the market exist in the internal management organization, they are the information which can be collected.

In summary, different methods of setting the weights can be shown in the following table:

Table 1: Different weight-setting methods

methods	Weight($W_{i,j}$) Transaction(1)	No-transaction(0)
0 weight	1	0
Full weight	1	1
Uniform weight	1	$\delta, \delta \in (0,1)$
The customer oriented weight	1	$\propto \sum_j X_{i,j}$
The product oriented weight	1	$\propto \left(m - \sum_j X_{i,j}\right)$
Real-time customer-oriented weight	1	$\propto \delta_i^{CR}$
Real-time product-oriented weight	1	$\propto \delta_j^{PL}$

4 Experiment and Results

The algorithm of OCCF based on real-time information model can be showed:
Input: customer-product matrix X, the rank r.
Output: the approximation matrix Y of X.

1) Initialize P with a random number less than 1;

2) Repeatedly using the formula (9), (10). update the U and P until the AUC value calculated has convergence;

3) $Y = UP$ return Y;

4) Using matrix Y make a recommend list.

This paper selected the area under the ROC curve (AUC) as a quality measure to compare measuring recommendation quality of different methods.

$$AUC = \frac{1}{|U|} \sum_u \frac{1}{|E(u)|} \sum_{(i,j) \in E(u)} \rho\left(\hat{x}_{ui} > \hat{x}_{uj}\right)$$

ρ, is an indicator function, it can be showed like:

$$\rho(\alpha) = \begin{bmatrix} 1 & \text{if } \alpha \text{ is true} \\ 0 & \text{else} \end{bmatrix}$$

$E(u)$, is the target pair of evaluation.

$$E(u) = \{(i,j)|\,(u,j) \in S_{test} \wedge (u,j) \notin (S_{test} \vee S_{train})\}$$

For all evaluations, we use the frequently-used datasets, the MovieLens dataset that GroupLens research group provided. The dataset contains 943 users and 1682 films; each user has at least 20 film scores. Thus, the total of assessment records is 100,000. Despite huge number, this data set is very sparse, sparsity is 6.305%, which is only 6.305% of the items are rated. The simulation environment includes: the PC with windows 7, Intel Core Duo processor and 4G RAM.

When we make the $r = 3$, Figure 1 is the typical results for all data sets, with each method running.

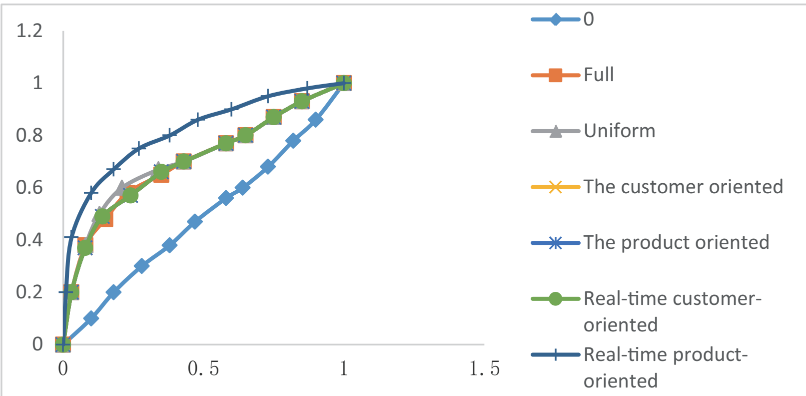

Figure 1: An AUC comparison that all methods in a single run.

Obviously 0 weight is the worst, its' the ROC curve almost shows the diagonal. In all other ways, the degree of improvement in performance is similar, we can learn from the shape of the ROC. In the new methods, the real-time product-oriented weight method's ROC curve is the best of all. Compare with other methods, the real-time customer-oriented weight method has some performance improvement.

Then we repeated 10 times by changing the rank and got the average value of AUC. Like table 2 and Fig 2 shows:

From the figure and table, we found that 0 weight methods performed badly, while the Real-time customer oriented method and Real-time product oriented method had good effect for sake of the recommend.

Table 2: The average AUC of different rank and methods

		rank					
		1	3	6	9	12	15
methods	0	0.51	0.501	0.505	0.497	0.494	0.49
	Full	0.71	0.699	0.697	0.693	0.689	0.687
	Uniform	0.699	0.712	0.71	0.708	0.705	0.702
	The customer oriented	0.692	0.701	0.698	0.696	0.692	0.693
	the product oriented	0.692	0.702	0.699	0.698	0.695	0.694
	Real-time customer-oriented	0.697	0.718	0.714	0.71	0.709	0.704
	Real-time product-oriented	0.779	0.78	0.778	0.78	0.781	0.779

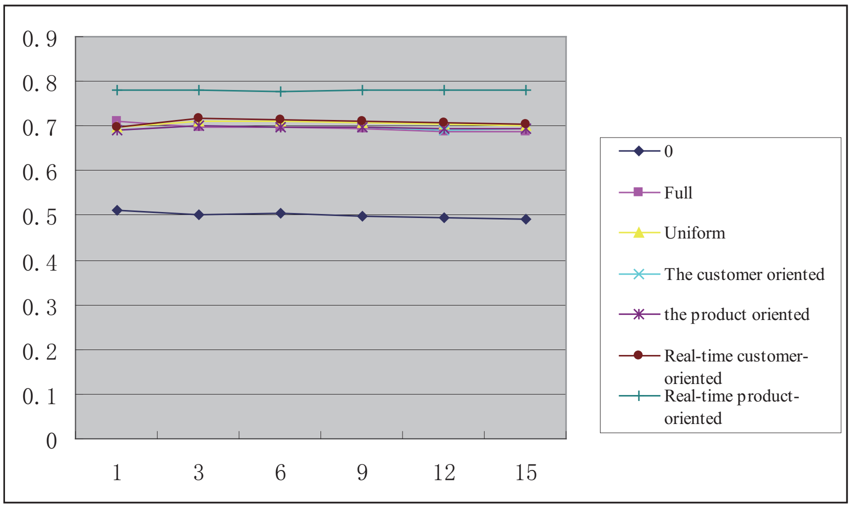

Figure 2: The average AUC of different ranks and methods.

5 Conclusion

On the basis of previous studies, this paper presents an integrated approach, through different weights to the one class collaborative filtering problem obtained the use of real-time information. The integration of real-time information including: the information of product launched into the market which is related to product life cycle, the information about product itself, and customer recently visiting information related to the customer relationship management. But this datasets used aren't from e-commerce websites, and it can't completely replace business process and actions. So how to apply the model to e-commerce needs to practice in the future. The next work is to test this model by changing the number of customers, products and transactions.

Now real-time information in the field of collaborative filtering experiments is limited, confined to movie ratings recommendations problem. One class collaborative filtering method which is based on e-commerce transaction record set, involved the customer's history access to sites and product information that launched into market, and got a good result in this paper. This solution to one class collaborative filtering problems will benefit the field of cross-selling e-commerce, personalized, targeted recommendations sales.

Bibliography

[1] Goldberg D., D. Nichols, B. M. Oki, and Terry D., Using collaborative filtering to weave an information tapestry, *Communications of ACM*, 35 (12): 61–70,1992.

[2] Li G. and Li L., Based on matrix decomposition algorithm for single-class collaborative filtering recommendation, *Application Research of Computers*, 29(5): 1662-1665, 2012.

[3] Shani G., D. Heckerman, and R. I. Brafman, An MDP-based recommender system, *Journal of Machine Learning Research*, 6: 1265–1295,2005.

[4] Banati H. and Mehta S., A Multi-Perspective Evaluation of ma and ga for collaborative Filtering Recommender System, *International J of Computer Science & Information Technology* 2(5): 103-22, 2010.

[5] Yang S. and Xue W., Classification based on a single collaborative filtering recommendation algorithm, *Computer Engineering* ,19(37): 59-61,2011.

[6] Pan R. and M. Scholz., Mind the gaps: Weighting the unknown in largescale one-class collaborative, *In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 667-676, 2009.

[7] Sindhwani V., S.S. Bucak, J. Hu, and A. Mojsilovic, A family of nonnegative

matrix factorizations for one-class collaborative filtering problems, *In Proc. of 3rd ACM RecSys Workshop on Recommendation-based Industrial Applications*, 2009.

[8] Koren Y., Collaborative filtering with temporal dynamics, *In Proc. of 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 447-456, 2009.

[9] Lu Z., D. Agarwal, and I. S. Dhillon, A spatio-temporal approach to collaborative filtering, *In Proc. of 3rd ACM RecSys Workshop on Recommendation-based Industrial Applications*.13-20, 2009.

[10] Xiang L. and Yang Q., Time-dependent models in collaborative filtering based recommender system, *In Proceedings of2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, (1), 450-457, 2009.

[11] Xiong L., X. Chen, T. Huang, J. Schneider, J. G. Carbonell, Temporal

collaborative filtering with Bayesian probabilistic tensor factorization, *In Proceedings of SIAM International Conference on Data Mining*, 2010.

[12] Srebro N. and Jakkola T., Weighted low rank approximations, Proceedings of 20th International Conference on Machine Learning, 720-727, 2003.

[13] Daniel D. Lee and H. Sebastian Seung, Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755): 788-791, 1999.

[14] Lan W. and Zhengjun Z., Collaborative filtering algorithm based on time weight, *Journal of computer application*, 27(9): 2302-2303, 2007.

[15] Huaizhen Y. Xiaoqi C. and Meilian L., Research on the Personalized Recommendation Algorithm Based on Time Weight, *Computer engineering& science*, 31(6): 126-128, 2009.

[16] Donghui L , Dewei P. and Hui Zh., Collaborative Filtering Algorithm Based on Time Weight and User's Feature, *Journal of wuhan university of technology*, 34(5): 144-148, 2012.

[17] Smaranda Cosma, Mădălina Văleanu, Dan Cosma, Dana Vasilescu, Grigor Moldovan, Efficient Data Organisation in Distributed Computer Systems using Data Warehouse, *INT J COMPUT COMMUN*, 8(3): 367-373, 2013.

[18] Kabir Golam and Hasin M. Ahsan Akhtar, Evaluation of customer oriented success factors in mobile commerce using fuzzy AHP, *Journal of Industrial Engineering and Management*, 4(2): 361-386, 2011.