Mining the Change of Customer Buying Behavior for Collaborative Recommendations

Yeong Bin Cho[1], Yoon Ho Cho[2], Soung Hie Kim[1]

[1] Graduate School of Management, Korea Advanced Institute of Science and Technology

e-mail : ybcho@kgsm.kaist.ac.kr, seekim@kgsm.kaist.ac.kr

[2] School of e-business, Kookmin University

861-1Jungnung-dong, Sungbuk-gu, Seoul 136-702, South Korea

e-mail : www4u@hanmail.net

Abstract

The preference of customers change as time goes by. The existing Collaborative Filtering (CF) techniques has no room for including this change yet, although these techniques have been known to be the most successful recommendation technique that has been used in a number of different applications.

In this study, we proposed a new methodology for enhancing the quality of recommendation using the customers' dynamic behaviors over time. The proposed methodology is applied to a large department store in Korea, compared to existing CF techniques. Some experiments on the real world data show that the proposed methodology provides higher quality recommendations than other CF techniques, especially better performance on heavy users.

keywords:

Recommendation systems, Dynamic Behaviors, Collaborative Filtering, SOM, Association Rules

1. Introduction

To date, a variety of recommendation techniques (Balabanović & Shoham, 1997; Basu, Hirsh & Cohen, 1998; Hill et al., 1995; Lawrence et al., 2001; Resnick et al., 1994; Sarwar et al., 2001; Shardanand & Maes, 1995) has been developed. Collaborative filtering has been known to be the most successful recommendation technique that has been used in a number of different applications such as recommending web pages, movies, articles and products (Hill et al., 1995; Resnick et al., 1994; Shardanand & Maes, 1995;Cho et al, 2002; Cho & Kim, 2004). Collaborative filtering identifies customers (neighbors) whose preferences are similar to those of a given customer and recommends products neighbors of a given customer have liked. The existing Collaborative filtering techniques are not possible to conduct a preference of customers change as time goes by.

Let us consider an example in Table 1. This example finds out the recommendation product for the CID011 customer using purchasing information from CID001 to CID010. A "+" indicates that the customer purchased the product and a "-" indicates that the customer did not purchase. To predict the product that CID011 customer is likely to buy using existing CF algorithm, we have to look for customers (neighbors) that have a similar purchasing pattern with CID011 customer. CID001 and CID003 become customers (neighbors) of CID011, since CID011 customer bought "Perfumes", "Skincares", "Dresses" and CID001 and CID003 bought same products. And then, it is occurred a conflict to recommend the product suitable for CID011 customer exactly, due to CID001 and CID003 bought different product: "Bags", "Shoes", respectively. In this case, it is difficult to search for recommending a product to CID011 customer and only have to select one of 4 alternatives: one of bags of CID001 and shoes of CID003, both of two, and none. Accordingly, accuracy of recommendation is possible to deteriorate. However, if it is known the past purchasing history of each customer, we can elaborate a recommendation. Table 2 shows the purchasing history of each customer in Table 1. CID001 customer bought an order of "Perfumes", "Skincares", "Dresses" and CID003 customer had the buying sequence of "Dresses", "Skincares", "Perfumes". Because the dynamic sequence of CID011 customer over time is like that of CID001, the recommendation product suitable for CID011 become "Bags". Additional information about dynamic behavior over time enables to enhance an accuracy of recommendation.

<Table 1> Purchasing Information without dynamic behaviors

CID	Perfumes	Skincares	Knits	Dresses	Mufflers	Watches	Bags	Shoes
001	+	+		+			+	
002			+	+		+		
003	+	+		+				+
004			+	+		+		
005					+		+	
006					+	+	+	
007					+		+	+
008			+		+		+	+
009			+		+			+
010			+	+				+
011	+	+		+				

<Table 2> Dynamic Purchasing Information over time

CID	T-3 period	T-2 period	T-1 period	T period
001	Perfumes	Skincares	Dresses	Bags
002	Watches	Dresses	-	Knits
003	Dresses	Skincares	Perfumes	Shoes
004	Watches	Dresses	-	Knits
005	Mufflers	-	Bags	-
006	-	Mufflers	Bags	Watches
007	Mufflers	-	Bags	Shoes
008	Knits	Mufflers	Bags	Shoes
009	Knits	Mufflers	-	Shoes
010	Knits	Dresses	-	Shoes
011	Perfumes	Skincares	Dresses	?

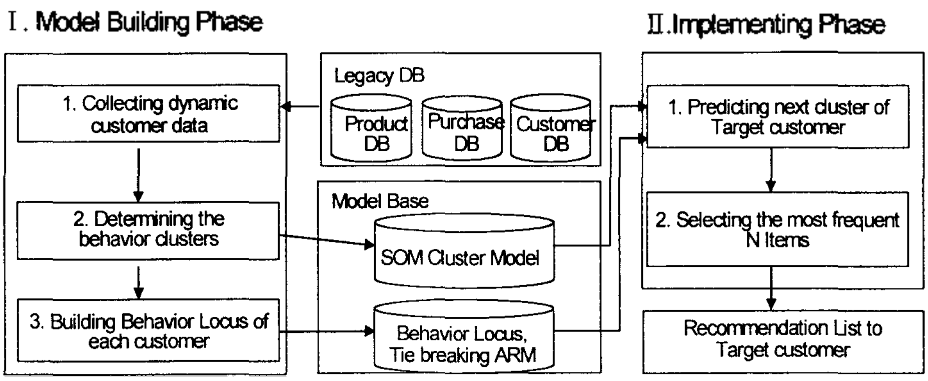
As shown this example, the dynamic behavior of customers with time may be another way enhancing accuracy of recommendations. However, traditional CF algorithms have no room for including these dynamic behaviors of customer. The proposed methodology bases their decisions by analyzing dynamic behavior patterns of the individual customer as well as the behavior of other customers over time.

In this paper, we use SOM technique (Kohonen 1990 ; Kohonen 1995 ; Kohonen et al. 1996 ; Simula et al. 1999) to know the dynamic behavior of customers. The SOM model is served as a clustering method that transforms customer transactions into the behavior locus of customer over time and stored in the Model base. Also, we use an association rule mining (Agrawal et al. 1993; Agrawal and Srikant 1994) to break ties among the behavior loci of customers. Recommendations for a specific customer are drawn from the top-N products based on the behavior locus of other customer. To solve the typical scalability problem of recommendation, we use product taxonomy as a dimensionality reduction technique. The proposed methodology is applied to a large department store in Korea to validate a performance, compared to existing CF techniques.

2. Proposed Methodology

2.1. Overall Approach

As shown in Figure 1, the overall procedure of the proposed methodology can be divided into two components called model-building phase and implementing phase. A model-building phase is performed once to create a reliable model for recommendation, whereas implementing phase is gone into action when target customer is chosen for marketing campaigns to recommend products which posses a highly likelihood to buy. The details are explained step by step from now on.



[Figure 1] Overall procedure

2.2. Model Building Phase

2.2.1 Collecting dynamic customer data

We shall assume that a product class set P is classified into n different subclasses (brands) and each subclass consists of subclasses at lower level.

$$\mathrm{P} = \{P_1, P_2, \ldots\ldots, P_n\} \tag{1}$$

After determining the time frame, we are able to make a continuation of collecting data for the dynamic customer behavior. In general, it is known that transformation into a bit vector composed of 0 and 1 is efficient (Mobasher et al., 2000). Then, we define input data vector as the follows.

[*Definition 1*] Customer transaction bit vector

Given a customer transaction $A_{j,T-k} \in \mathrm{A}$, we define the transaction as a bit vector:

$$\overline{A_{j,T-k}} = \left\langle P_1^{A_{j,T-k}}, P_2^{A_{j,T-k}}, \ldots\ldots P_n^{A_{j,T-k}} \right\rangle \quad j = 1,2,\ldots, m,\ k = 0,1,\ldots, l,\ l \geq 1 \tag{2}$$

where $P_i^{A_{j,T-k}} = \begin{cases} 1, & \text{if } P_i \in A_{j,T-k} \\ 0, & \text{otherwise} \end{cases}$ and $P_i^{A_{j,T-k}}$ is 1 if the customer i purchase j product class at $T-k$ period, otherwise 0.

2.2.2. Determining the behavior clusters

In this section, all transactions of customers in the database are clustered, transforming to customer transaction bit vectors based on prior purchase behavior. We use the SOM clustering method to assign the transactions of customers of a period into groups with similar patterns. This transaction-clustering enables to discover the dynamic behavior of a customer in the way which produces a behavior locus of a customer over time.

2.2.3. Building the behavior locus of each customer

The transaction clustering in previous section, assuming that the number of clusters is q, result in a set of clusters as follows:

$$\mathrm{C} = \{C_1, C_2, \ldots\ldots, C_q \tag{3}$$

where each C_i is a subset of $\overline{\mathrm{A}}$ in (2).

Since each cluster only represents a group of transactions with similar patterns at a different time frame, transaction clusters by themselves cannot capture a customer's dynamic behavior over time. Therefore, a rearrangement according to customer and time has need for detecting the dynamic behavior of each customer. The SOM model, visualized as a locus of each customer on the map, makes it easily possible to track the behavior dynamics. We can make the locus of customer behavior clusters by evaluating of which cluster the transaction data of each customer at each period belong to. This locus can find from a moving record of each customer in SOM map.

[Definition 2] The Locus of Customer Behavior

We define the Locus of customer behavior over time L_i as follows:

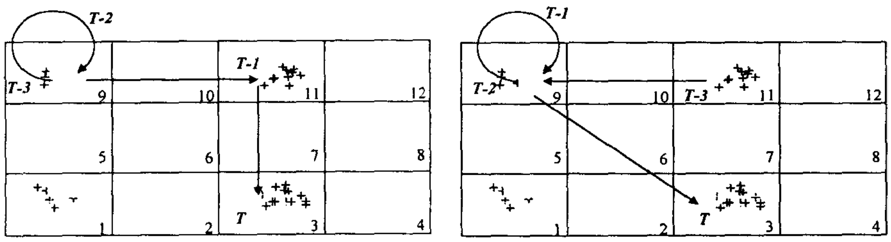
$$L_i = \{C_{i,T-l}, C_{i,T-l+1}, \ldots, C_{i,T-1}, C_{i,T}\},\quad i = 1,2,\ldots,m \tag{4}$$

where $C_{i,T-k} \in \mathrm{C}, k = 0,1,2,\ldots,l, l \ge 1$.

The Locus represents that a customer transaction clusters changes l times before T and goes to $C_{i,T}$ at T period. This process, searching for a locus, is simply done by running SOM model. The following example represents the sample locus of a customer behavior.

[Example1]

The behavior locus of CID 001, L_{001} is {9,9,11,3} and L_{003} is {11,9,9,3}, as shown Figure 2. L_{001} means that customer CID001 stayed $T-3$ period and moved in cluster 9, cluster 11, and reached in cluster 3 at T.



(a) CID 001 (b) CID 003

[Figure 2] The behavior loci of CID001 and CID 003

2.3. Implementing Phase

2.3.1. Predicting next cluster of target customer

Once the behavior clusters by SOM model have been computed, transaction data of target customer during l periods before T can be assigned to a matching cluster. We then determine which cluster would be provided to the target customer at T time as recommendations.

With the SOM model and the loci of customer stored in the Model Base, the locus prediction of target customer during l periods before T is conducted and the best matching locus in Model Base is searched for. The cluster locus of target customer which transformed by the SOM model is compared with the loci derived from training data, stored in the Model Base and then we find out the best matching locus. For doing this, the similarity measure can be represented by whether the cluster number of locus of the target customer during l time period are identical with the cluster number of the loci of the Model Base at the same time period, respectively. To formalize this concept, we define the following formula.

[Definition 3] Similarity measure

Let L^* be the locus of target customer and L_i be locus i in Model Base. And let us SM_i^* denote the similarity measure between L^* and L_i. SM_i^* is defined as follows:

$$SM_i^* = \sum_{k=1}^{l} S_{i,T-k}^* \tag{5}$$

where $S_{i,T-k}^* = \begin{cases} 1, & \text{if } C_{T-k}^* = C_{i,T-k}, C_{T-k}^* \in L^* \text{ and } C_{i,T-k} \in L_i \\ 0, & \text{otherwise} \end{cases}$

The above definition presents that if each cluster of locus of target customer is equal to that of locus i at same period, then $S_{i,T-k}^*$ is 1, otherwise 0. Note that L^* has only l clusters, not $l+1$, due to entered only purchasing data during past l periods into the SOM model. The higher SM_i^* value is, the more identical the locus of target customer is to the keeping loci.

However, if the length of l is not long enough, it may take place many ties on the similarity measure. It is difficult to choose that the rule is well suited for predicting the cluster of target customer at next period. And thus, another measure is requisite to break ties of the similarity measure. We introduce the tie breaking measure using the association technique. The frequent and confident loci of customer behaviors generated using association algorithm are stored in the Model Base for breaking the ties.

[Definition 4] Tie breaking measure

To break ties of loci i's having same maximum SM_i^*, let TB_i^j be the measure for calculating the similarity between the behavior loci i's and association rule j keeping in the Model base. Then, TB_i^j is defined as follows:

$$TB_i^j = \sum_{k=1}^{l} S_{i,T-k}^j \times Support_j \times Confidence_j \tag{6}$$

From the above definition, we can determine that the cluster of target customer at time T is a consequence part $r_{j,T}$ of the j association rule with maximum TB_j^i.

[Example 2]

Table 3 presents the behavior loci of 10 customers through the SOM model and Table 4 show the locus of target customer ID011. The maximum SM_i^* is $SM_{001}^{011} = 3$ and there is no tie of other customer's locus. Thus, the cluster of ID011 at time T is 3.

<Table 3> The behavior loci of 10 customers

CID	T-3	T-2	T-1	T
001	9	9	11	3
002	9	11	1	3
003	11	9	9	1
004	9	11	1	3
005	11	1	9	1
006	1	11	9	3
007	11	1	9	1
008	9	11	9	1
009	9	11	1	1
010	9	11	1	1

$$SM_i^* = \sum_{k=1}^{l} S_{i,T-k}^* \tag{5}$$

where $S_{i,T-k}^* = \begin{cases} 1, & \text{if } C_{T-k}^* = C_{i,T-k}, C_{T-k}^* \in L^* \text{ and } C_{i,T-k} \in L_i \\ 0, & \text{otherwise} \end{cases}$

The above definition presents that if each cluster of locus of target customer is equal to that of locus i at same period, then $S_{i,T-k}^*$ is 1, otherwise 0. Note that L^* has only l clusters, not $l+1$, due to entered only purchasing data during past l periods into the SOM model. The higher SM_i^* value is, the more identical the locus of target customer is to the keeping loci.

However, if the length of l is not long enough, it may take place many ties on the similarity measure. It is difficult to choose that the rule is well suited for predicting the cluster of target customer at next period. And thus, another measure is requisite to break ties of the similarity measure. We introduce the tie breaking measure using the association technique. The frequent and confident loci of customer behaviors generated using association algorithm are stored in the Model Base for breaking the ties.

[Definition 4] Tie breaking measure

To break ties of loci i's having same maximum SM_i^*, let TB_i^j be the measure for calculating the similarity between the behavior loci i's and association rule j keeping in the Model base. Then, TB_i^j is defined as follows:

$$TB_i^j = \sum_{k=1}^{l} S_{i,T-k}^j \times Support_j \times Confidence_j \tag{6}$$

From the above definition, we can determine that the cluster of target customer at time T is a consequence part $r_{j,T}$ of the j association rule with maximum TB_j^i.

[Example 2]

Table 3 presents the behavior loci of 10 customers through the SOM model and Table 4 show the locus of target customer ID011. The maximum SM_i^* is $SM_{001}^{011} = 3$ and there is no tie of other customer's locus. Thus, the cluster of ID011 at time T is 3.

<Table 3> The behavior loci of 10 customers

CID	T-3	T-2	T-1	T
001	9	9	11	3
002	9	11	1	3
003	11	9	9	1
004	9	11	1	3
005	11	1	9	1
006	1	11	9	3
007	11	1	9	1
008	9	11	9	1
009	9	11	1	1
010	9	11	1	1

<Table 4> The behavior locus of customer CID011

CID	T-3	T-2	T-1	T
011	9	9	11	?

2.3.2. Recommending the most frequent N items

The final step of the proposed methodology is to ultimately derive the top-N recommendation from the predicted cluster for target customer at time T. For each target customer, we produce a recommendation list of N products that the target customer is most likely to purchase. Recommendations for a specific target customer are drawn from the list of popular products in the assigned cluster for this customer.

Let us denote C^* the predicted cluster of target customer at T period, as determined in previous section. We can determine the top-N product recommendation list for target customer is the most frequently purchased products among products in the cluster.

[Definition 5] Recommendation List for target customer

Let us $MF(r_1)$ denotes the maximum purchased product at T period in the cluster such that products that the target customer has already purchased are excluded. $MF(r_2)$ is the next highest, and $MF(r_N)$ is the Nth highest. Then, Recommendation list for target customer is represented as $MF(r_1), MF(r_2),......, MF(r_N)$: while $MF(k)$ is computed as follows:

$$MF(k) = \sum_{j \in C^*} P_{ik}^{A_{j,T}} \times N_{ik}^{T} \tag{7}$$

where $P_{ik}^{A_{j,T-k}} = \begin{cases} 1, & \text{if } P_{ik} \in A_{j,T} \\ 0, & \text{otherwise} \end{cases}$, and N_{ik}^{T} is the number of P_{ik} product sold in T period, and P_{ik} is k leaf product in i product class.

3. Applications and Experiment

3.1. Data sets

We used real-world data to examine the performance of the proposed approach. The data used in the experiment is transaction records of woman goods of H department store, the third largest department store in Korea. We used transaction records obtained during the 8-month period from May to December 2000, in order to signify customer's behavioral characteristics over time. The input data from H department store database consists of transactions of 18,843, products of 557, and contains customer purchase data for 1,833 customers. Customers suitable for recommendation restricted loyal customer who have purchased frequently and recently, since it is difficult to discover a dynamic purchase behavior of rarely purchased customer over time. Time unit for analysis set on a month because there are rare customers who purchase a product daily or weekly in a department store. Interviews with domain expert indicated that a loyal customer may de defined by a customer who has at least purchased more than 1 times per month during consecutive 4 months. 310 customers fell into this category and the number of products they purchased was equal to the total number of products as mentioned previously.

3.2. Evaluation measures

To evaluate the quality of the recommendation set, recall and precision have been widely used in field of recommender systems (Basu et al., 1998; Billsus & Pazzani, 1998; Lin et al., 2000, 2002; Sarwar et al., 2000). These are computed as follow:

$$\text{Precision} = \frac{\text{Number of hit products}}{\text{Total number of recommended products}} \tag{8}$$

$$\text{Recall} = \frac{\text{Number of hit products}}{\text{Total number of purchased products at T period}} \tag{9}$$

$$\text{F1} - \text{measure} = \frac{\text{Recall} \times \text{Precision}}{(\text{Recall} + \text{Precision})/2} \tag{10}$$

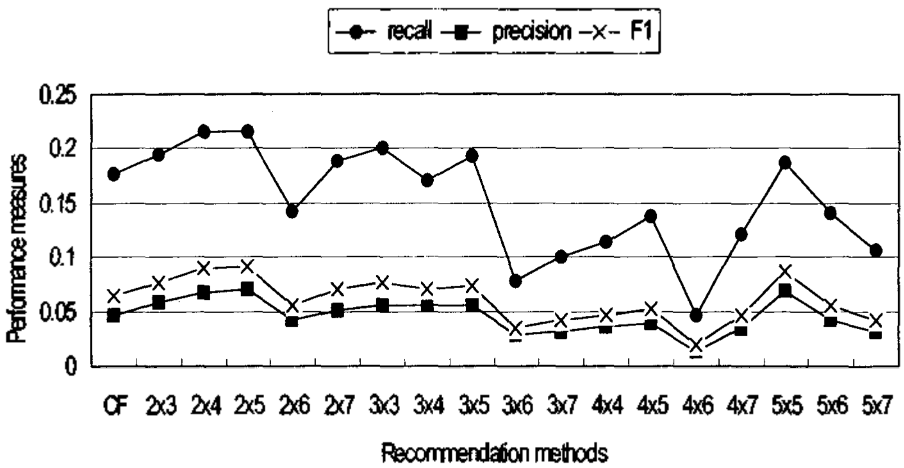
3.3. Results and discussions

We performed an experiment where we varied the number of clusters to determine the effectiveness of the recommendations by computing three metrics as mentioned before. Also, we compared the recommendation accuracy of our proposed methodology with that of the benchmark CF algorithm with the best accuracy. Fig. 3 shows our experimental results. For all three measures, cases of the number of cluster 6(2×3), 8(2×4), 9(3×3), 10(2×5), 14(2×7), 15(3×5), 25(5×5) were higher than the benchmark CF algorithm, on the contrary, 12(2×6, 3×4), 16(4×4), 18(3×6), 20(4×5), 21(3×7), 24(4×6), 28(4×7), 30(5×6) were lower. Only recall of 12(3×4) indicate the lower value compared to the CF, while the rest two measures are not.

We cannot find out a general tendency according to the number of clusters, for example, the accuracy become higher if the number of clusters increase and vice versa. Looking into the results, we can describe that the number of clusters does affect the accuracy of top-N recommendations. Intuitively, these are fairly reasonable because all existing cluster methods have suffered from the optimal number of clusters and then it is still a challenging problem (Nour and Madey, 1996). These results are depended on to what extent each SOM model takes the locus of target customer into account. To prevent from an over-fitting of clustering, it is necessary to determine the number of SOM clusters to account for the locus of customer exactly.

A level of minimum support and minimum confidence didn't greatly influence on finding the fittest association rule for the locus of target customer, since the fitness measure proposed in this study weaken the effect of minimum value of support and confidence.

F1 and precision of 8(2×4), 10(2×5), 25(5×5) clusters have statistical significance with $p<0.05$, when using two-tailed t-test. The proposed methodology works better performance than the traditional CF technique, if the numbers of SOM clusters are chosen well. The proposed methodology that uses the optimal choice for the number of cluster, in case of 10(2×5), works even better, achieving an average improvement of 40% and 52%, F1 and precision, respectively.



[Figure3] Comparison of CF and the proposed methodology by the number of clusters

4. Conclusion

The preference of customers will change as time goes by. In this study, we describe a model-based approach for mining the change of customer buying behavior over time, and discuss solutions to the problems of data preprocessing, behavior locus extraction, and making recommendations based on the extracted locus. Using the derived recommendation list, company may be able to perform effective one-to-one marketing campaigns of providing individual target customer with a personalized product recommendation.

The research work presented in this paper makes several contributions to the recommender systems related research. First of all, we applied the change of preferences over time to improving the accuracy of the recommendation. Additionally, we can find out that the proposed methodology is more suitable for a heavy user. Second, we developed the technique to capture implicit ratings by tracking customers' shopping behaviors rather than only collecting explicit ratings, thereby reducing the sparsity.

There are some possible extensions to this work. From result of this study, we knew which product target customer is likely to buy, but we have not known yet what time the customer is willing to buy. Another research for analyzing customers' past purchasing pattern will enable to detect an appropriate time for recommendation. Also, since the accuracy of all model-based approaches is deteriorated as time passed, the model has to be dynamically updated to reflect the user's evolving interests over time. It needs a repair plan to see how the predictive capabilities of a model decrease as time increase. Furthermore, it will be an interesting research area to conduct a real marketing campaign to target customers using our methodology and then to evaluate its performance.

References

[1] Agrawal, R., Imielinski, T. & Swami, A. (1993). Mining association rules between sets of items in large databases. Proceedings of the ACM SIGMOD Conference on Management of Data, 207-216.

[2] Agrawal, R. & Srikant, R. (1994). Fast algorithm for mining association rules. Proceedings of the International Conference on Very Large Databases (VLDB-94), 487-499.

[3] Balabanović, M. & Shoham, Y. (1997). Content-based, collaborative recommendation. *Communacations of the ACM, 40* (3), 66-72.

[4] Basu, C., Hirsh, H., & Cohen, W. (1998). Recommendation as classification: using social and content-based information in recommendation. In *Proceedings of the 1998 Workshop on Recommender Systems*, 11-15, AAAI Press.

[5] Billsus, D., & Pazzani, M. J. (1998). Learning collaborative information filters. Proceedings of the 15th International Conference on Machine Learning, 46–54.

[6] Cho, Y.H., Kim, J,K., and Kim, S.H. (2002). A Personalized Recommender System based on Web Usage Mining and Decision Tree Induction. Expert Systems With Applications, Vol. 23(3),329-342.

[7] Cho, Y.H. & Kim, J,K. (2004). Application of Web usage mining and product taxonomy to collaborative recommendations in e-commerce. Expert Systems With Applications, Vol. 26(3), 234-246

[8] Hill, W., Stead, L., Rosenstein, M., & Furnas, G. (1995). Recommending nad Evaluating Choices in a Communication of Use. In proceedings of CHI 95.

[9] Kohonen, T. (1990). The Self-Organizing Map. Proceedings of the IEEE 78(9): 1464-1480.

[10] Kohonen, T. (1995). Self-Organizing and Associative Memory. Berlin: Springer-Verlag.

[11] Kohonen, T., Oja, E., Simula, O., Visa, A. & Kangas, J. (1996). Engineering applications of the

Self-Organizing Map. Proceedings of the IEEE 84(10): 1358-1384.

[12] Lawrence, R.D., Almasi, G.S., Kotlyar, V., Viveros, M.S., & Duri, S.S.(2001). Personalization of Supermarket Product Recommendation, Data mining and Knowledge Discovery, 5(1-2), 11-32.

[13] Lin, W., Alvarez, S. A., & Ruiz, C. (2000). Collaborative recommendation via adaptive association rule mining. Proceedings of the WEBKDD.

[14] Lin, W., Alvarez, S.A., & Ruiz C. (2002). Efficient Adaptive-Support Association Rule Mining for Recommender Systems. Data Mining and Knowledge Discovery, 6(1) 83-105.

[15] Nour, M. A. & Madey, G. R. (1996). Heuristic and optimization approaches to extending the Kohonen self organizing algorithm, European Journal of Operational Research, 93(2), 428-448

[16] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P. and Riedl, J. (1994). Grouplens: An open architecture for collaborative filtering of netnews. In Proceedings of the ACM 1994 Conference on Computer Supported Cooperative Work, 175-186.

[17] Sarwar, B., Karypis, G., Konstan, J. and Riedl, J. (2000). Analysis of Recommendation Algorithms for E-commerce. In Proceedings of ACM E-commerce 2000 conference, 15-167.

[18] Shardanand, U. and Maes, P. (1995). Social Information Filtering: Algorithms for Automating 'Word of Mouth'. In proceedings of CHI 95.

[19] Simula, O., Vesanto, J., Alhoniemi, E. & Hollmen, J. (1999). Neuro-Fuzzy Techniques for Intelligent Information Systems. Springer Verlag.