한국지능정보시스템학회논문지 제9권 제3호 2003년 12월(pp.177~191)

Development of a Personalized Recommendation Procedure Based on Data Mining Techniques for Internet Shopping Malls

Jae Kyeong Kim
School of Business Administration,
Kyung Hee University
(jaek@khu.ac.kr)

Do Hyun Ahn
School of Business Administration,
Kyung Hee University
(adh@khu.ac.kr)

Yoon Ho Cho
Department of Internet Information,
Dongyang Technical College
(yhcho@dongyang.ac.kr)

Recommender systems are a personalized information filtering technology to help customers find the products they would like to purchase. Collaborative filtering is the most successful recommendation technology. Web usage mining and clustering analysis are widely used in the recommendation field. In this paper, we propose several hybrid collaborative filtering-based recommender procedures to address the effect of web usage mining and cluster analysis. Through the experiment with real e-commerce data, it is found that collaborative filtering using web log data can perform recommendation tasks effectively, but using cluster analysis can perform efficiently.

Key Words : Product recommendation, Web usage mining, Clustering, Collaborative Filtering, Personalization

논문접수일 : 2003년 10월 계재확정일 : 2003년 12월 교신저자 : 김재경

1. Introduction

The rapid expansion of e-commerce forces existing recommender systems to deal with a large number of customers and products [11]. Collaborative Filtering (CF) [4], [8], [14] has been known to be the most successful recommendation technique that has been used in a number of different applications. However, CF based recommender systems suffer from two fundamental problems, sparsity and scalability. To overcome these problems, we propose to use web usage mining and cluster alalysis. We designed two research questions, and four procedures to show experimental results of their performance.

The characteristics of our suggested procedures are as follows: (1) Cluster analysis for customer segmentation is applied to improve scalability of recommender systems, (2) Products are recommended to target customers according to

* 본 연구는 한국학술진흥재단 2002년도 선도연구자지원에 의하여 이루어졌음(KRF-2002-041-B00167).

Web usage mining based CF to address sparsity issue. To compare the effect of clustering and web usage mining, the procedures are evaluated with real Internet shopping mall data.

The remainder of this paper is organized as follows. Section 2 reviews related research works, and section 3 provides our research framework. Section 4 describes experimental works, and conclusions and future works are provided at section 5.

2. Backgrounds

2.1 Recommender Systems

Recommender systems are changing the face of e-commerce on the Internet by enabling web sites to help their customers find products they will be interested in buying. These systems apply data analysis techniques to the problem of helping customers find the products they would like to purchase at e-commerce sites by producing a prediction score or a list of top-N recommended products for a given customer. For instance, a recommender system of Amazon.com (www. amazon. com) suggests books to customers based on other books the customers have told Amazon they like. Recommendations are suggested based on demographics of the customers, overall top selling products, or past buying habit of customers as a predictor of future products [16]. In essence, these techniques try to personalize the e-commerce space for the customers. Among the different approaches applied to achieve personalization in e-commerce, CF is known to be arguably the most successful technique deployed in commercial applications as well as in academic research [6].

Recommender systems increase e-commerce sales in three ways [17]. First, recommender systems help to convert browsers into buyers by providing personalized recommendations on a variety of products. Second, recommender systems improve cross-sell by suggesting additional products for the customer to purchase. Third, recommender systems improve loyalty by creating a value-added relationship between the e-commerce site and the customer. Numerous recommender systems have been built for both research and practice.

2.2 Collaborative Filtering Algorithm

CF presents an alternative information evaluation approach based on the judgments of human beings. It attempts to automate the "word of mouth" recommendations that we regularly receive from family, friends, and colleagues. This inclusiveness circumvents the scalability problems and it becomes possible to review millions of books [16]. One of automated CF systems uses a machine learning approach called the nearest neighbor algorithm to provide a computer implementation of this technique. Such systems maintain a database containing the ratings that each user has given to each item that each user has evaluated (e.g. in the form of a score from 1

to 5). For each user in the system, the recommendation engine computes a neighborhood of other users with similar options; this neighborhood is usually based on a proximity measure such as correlation or cosign. To evaluate other items for this user, the system forms a normalized and weighted average of the opinions of the user's neighbors. Several recommender systems have been developed based on automated CF.

2.3 Cluster Analysis

Cluster analysis has been applied to a wide range of disciplines such as data mining, statistics, machine learning, spatial database technology, biology, and marketing. Owing to the huge amount of data collected in databases, cluster analysis has recently become a highly active topic [7]. Cluster analysis has been studied extensively for many years, focusing mainly on distance-based cluster analysis such as k-menas, k-medoids, and several other methods [1]. In data mining, efforts have focused on finding methods for efficient and effective cluster analysis in large databases [2]. Active themes of research focus on the scalability of clustering methods, the effectiveness of methods for clustering complex shapes and types of data, high-dimensional clustering techniques, and methods for clustering mixed numerical and categorical data in large databases. Cluster analysis has been applied to the recommendation field [2], [3], [9]. Earlier collaborative filtering research conducted in the Usenet domain [9] reported the benefits of partitioning. In particular, they found improved prediction quality with partitioned newsgroups compared to the whole Usenet. Nowadays many current collaborative filtering methods use cluster analysis for the formation of neighborhood [19]. In this paper, we applied cluster analysis to CF to improve the quality of recommendations, especially to solve scalability issue of traditional CF systems.

2.4 Web Usage Mining

Web usage mining is the process of applying data mining techniques to the discovery of behavior patterns based on web log data [5], [13], [18]. In the advance of e-commerce, the importance of Web usage mining grows larger than before. The overall process on Web usage mining is generally divided into two main tasks; data preprocessing and pattern discovery. Mining behavior patterns from Web log data needs the data preprocessing tasks that include data cleansing, user identification, session identification, and path completion. Data cleansing performs merging Web logs from multiple servers, removing irrelevant and redundant log entries with filename suffixes such as gif, jpeg, map, count, cgi, and so on, and parsing of the logs. To track individual user's behaviors at a Web site, user identification and session identification is required. For Web sites using session tracking such as URL rewriting, persistent cookies or embedded session IDs, user and session identification is trivial. Web sites without session tracking must rely on

heuristics. Path completion may also be necessary because of local or proxy level caching. Cooley et al. presented a detailed description of data preprocessing methods for mining Web browsing patterns [5]. The pattern discovery tasks involve the discovery of association rules, sequential patterns, usage clusters, page clusters, user classifications or any other pattern discovery method. Usage patterns extracted from Web data can be applied to a wide range of applications such as Web personalization, system improvement, site modification, business intelligence discovery, usage characterization, and so on [18].

There have been several customer behavior models for e-commerce. Menascé et al. have presented a state transition graph, called Customer Behavior Model Graph (CBMG) to describe the behavior of groups of customers who exhibit similar navigational patterns [12]. VandeMeer et al. have developed a user navigation model designed for supporting and tracking dynamic user behavior in online personalization [20]. The model supports the notion of a product catalog, user navigation over this catalog and dynamic content delivery. Lee et al. have provided a detailed case study of clickstream analysis from an online retail store [10]. They have analyzed the shopping behavior of customers according to the following four shopping steps; product impression, click-through, basket placement, and purchase. And they have applied micro-conversion rates (e.g., click-to-buy rate) computed for each adjacent pair of these steps in order to measure the effectiveness of efforts in merchandising.

3. Methodology

3.1. Research Design

In this section, we first set up the research questions that we examine in this study. Then, we suggest overall procedures designed to answer our research questions. We pose two research questions that will accomplish our research objective aforementioned:

Q1. Does clustering-based CF give better performance than CF alone?

We tested whether clustering-based CF improves the recommendation quality or not. Also it is tested whether clustering-based CF reduces the search space, so solves the scalability issue of traditional CF algorithm.

Q2. Does CF with web log data give better performance than CF with purchase data only?

Web usage mining is difficult and a time consuming process, but web log data contains more information about the behavior of customers than purchase data only. We tested whether web usage mining based CF improves the recommendation quality or not, and furthermore how much the quality difference is. Web usage mining based CF algorithm is tested also whether it solves sparsity issue or not. In order to test our research questions, we suggest the following

four different procedures:

Method 1: CF with Purchase data + No Clustering
Method 2: CF with Web log data + No Clustering
Method 3: CF with Purchase data + Clustering
Method 4: CF with Web log data + Clustering

3.2. Clustering Phase

We consider the application of clustering analysis to improve scalability of recommender systems. Konstan et al. indicate the benefits of applying clustering in recommender systems [1], [9]. The idea is to segment the customer of collaborative filtering system using a clustering algorithm and to use the segment for forming neighborhoods of CF phase. Formally, the customer set C is segment in C_1, C_2,$\cdots$, C_p, where $C_i \cap C_j = \emptyset$, for $1 \le i, j \le P$; and $C_1 \cup C_2, \cdots, C_j = C$. We first segment the cutomers using *k-means* algorithm and to use the segments for forming neighborhoods of CF phase. *K-means* algorithm has been known to be effective in producing good clustering results for many practical applications [1], [9]. The following is basic process of *k-means* algorithm. First, it randomly selects k of the customers, each of which inintially represents cluster mean or center. For each of the remaining customers, a customer is assighed to the cluster to which it is the most similar, based on the distance between the customer and the cluster mean. It then computes the new mean for each cluster. This process iterates until the criterion function converges.

The data used in recommendations are usually divided into three types: demographic data, behavior data, and psychographic data [15]. But in our research demographic data and behavior data are used, because psychographic data are difficult to be collected systematically from the e-commerce site.

3.3. CF Phase

A CF algorithm is composed of profile creation, neighborhood formation, and generation of recommended products.

Step 1. Profile Creation

A profile is a collection of information that describes a user. One of the important issues in the profile creation is what information should be included in a user profile.

Profiles based on Purchase data. Profiles based on purchase data are collections of historical purchasing transaction of n customer on m products. It is usually represented as an m x n customer-rating matrix P, such that $r_{i,j}$ is one if the ith customer has purchased the jth product, and zero, otherwise [16].

Profiles based on Web data. Profile based on Web data is constructed based on the following three general shopping steps in Web retailer:

1. *click-through*: the click on the hyperlink and the view of the Web page of the product,
2. *basket placement*: the placement of the product in the shopping basket,

3. *purchase*: the purchase of the product - completion of a transaction.

A basic idea of measuring the customer's preference is simple and straightforward. The customer's preference is measured by counting only the number of occurrence of URLs mapped to the product from clickstream of the customer. In general Internet shopping malls, products are purchased in accordance with above three sequential shopping steps, so we can classify all products into four product groups such as purchased products, products placed in the basket, products clicked through, and the other products. This classification provides an *is-a* relation between different groups such that purchased products *is-a* products placed in the basket, and products placed in the basket *is-a* products clicked through. From this relation, it is reasonable to obtain a preference order between products such that {products never clicked} π {products only clicked through} π {products only placed in the basket} π {purchased products}. Hence, it makes sense to assign the higher weight to occurrences of purchased products than those of products only placed in the basket. Similarly, the higher weight is given to products placed in the basket than those of products only clicked through, and so on.

Let p_{ij}^{c} be the total number of occurrences of click-throughs of a customer i across every products in a grain product class j. Likewise, p_{ij}^{b} and p_{ij}^{p} are defined as the total number of occurrences of basket placements and purchases of a customer i for a product j, respectively. p_{ij}^{c}, p_{ij}^{b} and p_{ij}^{p} are calculated from clickstream data as the sum over the given time period, and so reflect individual customer's behaviors in the corresponding shopping process over multiple shopping visits.

From the above terminology, we define the customer profile as the matrix of ratings $P = (P_{ij})$, $i = 1, \wedge$, M(total number of customers), $j = 1, \wedge$, N(total number of products), as follows:

$$p_{ij} = \left(\frac{p_{ij}^{c} - \min_{1 \le j \le N}(p_{ij}^{c})}{\max_{1 \le j \le N}(p_{ij}^{c}) - \min_{1 \le j \le N}(p_{ij}^{c})} + \frac{p_{ij}^{b} - \min_{1 \le j \le N}(p_{ij}^{b})}{\max_{1 \le j \le N}(p_{ij}^{b}) - \min_{1 \le j \le N}(p_{ij}^{b})} + \frac{p_{ij}^{p} - \min_{1 \le j \le N}(p_{ij}^{p})}{\max_{1 \le j \le N}(p_{ij}^{p}) - \min_{1 \le j \le N}(p_{ij}^{p})} \right) \times \frac{1}{3} \quad (1)$$

The p_{ij} range from 0 to 1, where more preferred product result in bigger value. Please note that the weights for each shopping step are not the same although they look equal as in Equation (1). From a casual fact that customers who purchased a specific product had already not only clicked several Web pages related to it but placed it in the shopping basket, we can see that Equation (1) reflects the weight difference.

Step 2: Neighborhood Formation

This step performs computing the similarity between customers and, based on that, forming a proximity-based neighborhood between a target customer and a number of like-minded customers. The process follows the same manner as that of

typical nearest-neighbor algorithms [17] except forming the neighborhood in the same customer segment. The details of the neighborhood formation are as follows.

Given the customer profile matrix P, the similarity between two customers a and b which is contained in customer segment C_i, denoted by $sim(a, b)$, is usually measured using either the *correlation* or the *cosine* measure. In our research, we use the following *correlation* measure. The similarity between two customers a and b is measured by calculating the *Pearson-r correlation*, which is given by

$$sim(a,b) = corr_{ab} = \frac{\sum_j (p_{aj} - \bar{p}_a)(p_{bj} - \bar{p}_b)}{\sqrt{\sum_j (p_{aj} - \bar{p}_a)^2 \sum_j (p_{bj} - \bar{p}_b)^2}} \quad (2)$$

where p_{aj} and p_{bj} are customer a and b's ratings on product j, respectively, and $\bar{p}_a$ and $\bar{p}_b$ are customer a and b's average ratings on all products, respectively.

Step 3: Generation of Recommendation list

This step is to ultimately derive the top-N recommendation from the neighborhood of customers. For each customer c, we produce a recommendation list of N products that the target customer is most likely to purchase. Previously bought products are excluded from the recommendation list in order to broaden each customer's purchase patterns or coverage. We suggest two different techniques for generating a recommendation list for a given customer.

Recommendation of the Most Frequently Purchased Product (MFP). This technique looks into the neighborhood and for each neighbor, scans through a sales database and counts the purchase frequency of the products. After all neighbors are accounted for, the system sorts the products according to their frequency count and returns the most frequently purchased products as the recommendation list. This technique assumes that the more a product is sold, the more popular it becomes. We apply this technique to CF with purchase data.

Recommendation of the Most Frequently Referred Product (MFR). Unlike MFP technique based on purchase frequencies of all neighbors, this technique sorts the products according to their reference frequencies. The reference frequency of the neighborhood of a particular customer a for a product j, $RF_{a,j}$, is defined below:

$$RF_{a,j} = \sum_{i \in \text{neighbors of customer } a} \left(\frac{r_{ij}^c - \min_{1 \le j \le N}(r_{ij}^c)}{\max_{1 \le j \le N}(r_{ij}^c) - \min_{1 \le j \le N}(r_{ij}^c)} + \frac{r_{ij}^b - \min_{1 \le j \le N}(r_{ij}^b)}{\max_{1 \le j \le N}(r_{ij}^b) - \min_{1 \le j \le N}(r_{ij}^b)} + \frac{r_{ij}^p - \min_{1 \le j \le N}(r_{ij}^p)}{\max_{1 \le j \le N}(r_{ij}^p) - \min_{1 \le j \le N}(r_{ij}^p)} \right| \quad (3)$$

where n is the number of products, and r_{ij}^c, r_{ij}^b and r_{ij}^p is the total number of occurrences of click-throughs, basket placements and purchases of a customer i for a product j, respectively. This method follows from the hypothesis that the more a product is referred, the higher the possibility of product's purchase becomes. The reference frequency is computed using clickstream data as in building the customer profile. We apply this technique to CF with web log data.

4. Experimental Evaluation

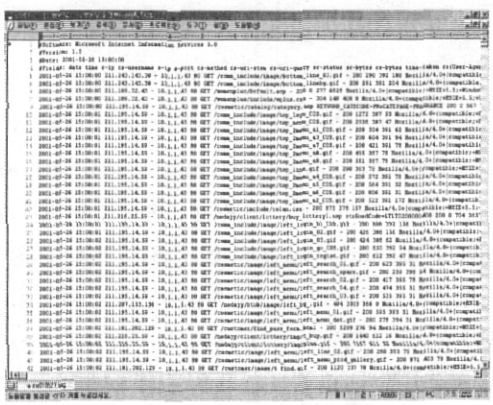
4.1. Data Preparation

For our experiments, we use Web log data and product data from the S Internet shopping mall that sells women's supplies.

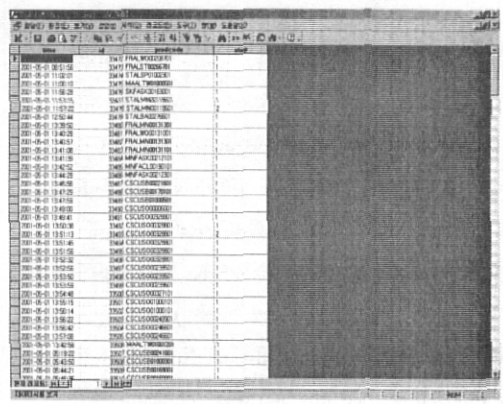
Web log data. The 110 log files was collected from four IIS Web servers during period between 1st May 2001 and 7th June 2001. The total size of log files is about 25,360MB, and total number of HTTP requests is about 510,000,000,000. For an application to our experiments, the preprocessing tasks such as data cleansing, user identification, session identification, path completion, and URL parsing were applied to the log files. Finally, we obtained a transaction database in the form of *<time, customer-id, product-id, shopping-step>* which the shopping-step represents one of the click-through step, the basket-placement step and the purchase step. This database contains transactions of 49597 customers on 278 products. In total, the database contains 428,510 records that consist of 781 purchase records, 5,350 basket-placement records, and 422,379 click-through records. <Figure 1> provides raw Web log data and the corresponding transaction database.

We set the period between 1st May 2001 and 24th May 2001 and the period between 25th May 2001 and 7th June 2001 as the training period and the test period, respectively. And then, as the target customers, we selected 130 customers who have purchased one more products in the training period and clicked one more products for the test period. Finally, the training set consists of 6,331 transaction records created by the target customers for the training period, and the test set consists of 677 click-through records created by them for the test period.

Product data. S Internet shopping mall deals with 7513 products. Table 1 shows products managed in S Internet shopping mall.



(a) Raw web log data



(b) Transaction database

<Figure 1> Web log preprocessing

<Table 1> example of product data set

prodcode	prodmane	classcode	classname
MWCACD00H02901	니트점퍼형가디건	MWCACD	가디건
MWCACT99H64701	라이트다운코트	MWCACT	코트
MWCACT99H85801	맨스노우롱다운	MWCACT	코트
MWCAJK00B73501	크링클쟈켓	MWCAJK	자켓
MWCAJK00B73601	스웨이드쟈켓	MWCAJK	자켓
MWCAJP00A42001	폴리버시블점퍼	MWCAJP	점퍼
MWCAJP00A55401	남성양면점퍼	MWCAJP	점퍼
MWCAJP00B37001	힙합후드점퍼	MWCAJP	점퍼
MWCASH00A41501	서큐넥버튼셔츠	MWCASH	셔츠
MWCASH00A41510	서큐넥버튼셔츠	MWCASH	셔츠
MWSUJK00F93801	3버튼쟈켓	MWSUJK	자켓
MWSUJK00F93901	기획정장자켓	MWSUJK	자켓
MWCASW00A46903	라운드스웨터	MWCASW	스웨터
MWCASWOOA46914	라운드스웨터	MWCASW	스웨터

4.2. Evaluation Metrics

Recommender systems research has used a number of different measures for evaluating the success of a recommender system. Main research objective of this paper is to develop new procedures for making recommendations that has better quality and more speed compared to previously studied approaches. Therefore, two evaluation metrics are employed for evaluating our procedures in terms of quality and performance requirements.

4.2.1. Quality evaluation metric

With the training set and the test set, our 4 methods work on the training set first, and then it generates a set of recommended products, called recommendation set, for a given customer. To evaluate the quality of the recommendation set, *recall* and *precision* have been widely used in the recommender system community [16]. Recall is defined as the ratio of the number of products in both test set and recommendation set to the number of products in test set.

Precision is defined as the ratio of the number of products in both test set and recommendation set to the number of products in recommendation set. Recall means how many of all the products in the actual customer purchase list are recommended correctly whereas precision means how many of the recommended products belong to actual customer purchase list. These measures are simple to compute and intuitively appealing, but they are often in conflict since increasing the size of recommendation set tends to increase recall but at the same time decrease precision [16]. Hence, a widely used combination metric called *F1 metric* that gives equal weight to

both recall and precision is employed for our evaluation, and computed as follows:

$$F1 = \frac{2 \times recall \times precision}{recall + precision} \quad (4)$$

4.2.2. Performance evaluation metric

To evaluate the scalability issue, we use a performance evaluation metric in addition to the quality evaluation metric. The *response time* are employed to measure the system performance. The response time defines the amount of time required to compute all the recommendations for the training set per second.

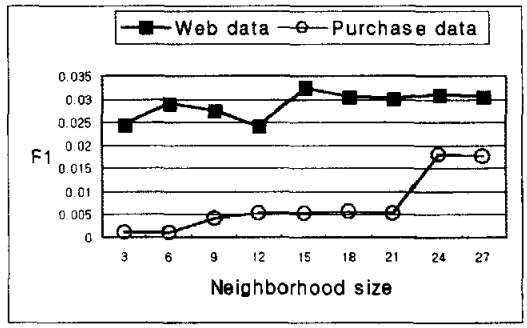
4.3. Experiment Results

In this section, we present a detailed experimental evaluation of the different procedures.

4.3.1. Experiments with neighborhood size

The size of the neighborhood has significant impact on the recommendation quality [16]. To determine the sensitivity of neighborhood size, we performed an experiment in which we varied the number of neighbors and computed the corresponding *F1* metric. <Figure 2> shows our experimental results. Looking into the results, we can see that the size of the neighborhood does affect the quality of *top-N* recommendations.

In general, the quality increases as we increase the number of neighbors, but, after a certain peak, the improvement gains diminish and the quality becomes worse. This reason may be that choosing too many neighbors result in too much noise for those who have high correlates. In the case of Web data, the peak is reached in the 15, whereas in case of Purchase data is reached in the 24. Hence, we used a neighborhood of size 15 for the Web data and that of 24 for the Purchase data as our ideal choice of neighborhood size.



<Figure 2> Impact of neighborhood size on recommendation quality

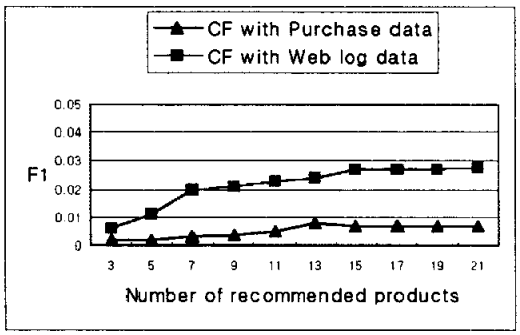
4.3.2. Effect of Web log data

Given the optimal values of the parameters, we compare CF with Purchase data with CF with Web log data. Our results are shown in <Figure 3>. It can be observed from the chart that CF with Web log data works better than CF with Purchase at all the number of recommended products. The recommendation with Web log data results better performance than that of Purchase data only.

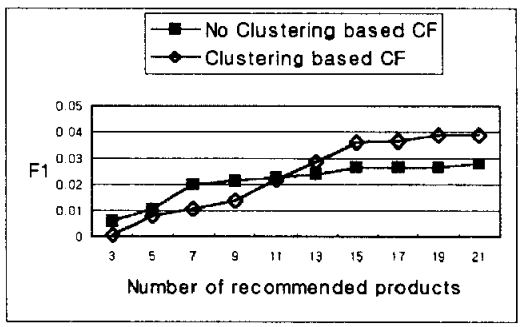
4.3.3. Effect of Clustering Analysis

We also compare No Clustering based CF

with Clustering based CF. Our results are shown in <Figure 4>. We can see that the quality of Clustering based CF is better than that of No Clustering based CF. However, using the Clustering technique is not robust performance, especially at a few number of recommended products.



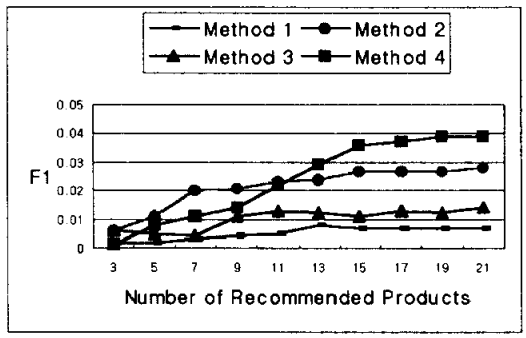
<Figure 3> Effect of Web log data



<Figure 4> Effect of Clustering Technique

4.3.4. Comparison of four Methods

With the number of recommended products from 3 to 21, <Figure 5> shows the comparison of method 1, 2, 3 and 4. It can be observed from the charts that Method 2 and Method 4 work better than Method 1 and Method 3 at all the number of recommended products. This implies that Web usage mining gives better results. Furthermore, we can see that the quality of Clustering based CF is better than that of No Clustering based CF. However, the application of clustering did not always give better performance.



<Figure 5> Comparison of four Methods

4.3.5. Performance comparison of hybrid algorithms

To compare the performance of our hybrid procedures with that of the benchmark CF algorithm, we performed an experiment in which we measure the *response time* of each procedure. The response time means the amount of time required to compute all the recommendations for the training set per second. <Table 2> shows the response time provided by the three algorithms. Looking into the results shown in <Table 2>, we can see that the performance of [Method 3, Method 4] is better than that of other methods. We believe this is due to the effect of Clustering technique.

<Table 2> Performance comparison of our hybrid algorithms

Hybrid Procedures	Response time (sec.)
Method 1: CF with Purchase data + No Clustering	405.2
Method 2: CF with Web log data + No Clustering	1173.5
Method 3: CF with Purchase data + Clustering	23
Method 4: CF with Web log data + Clustering	84

5. Conclusion

5.1. Summary

We suggested hybrid recommender procedures based on web usage mining, clustering and collaborative filtering. We experimentally evaluated our hybrid procedures on real e-commerce data and compared the effect of each approach.

Based on the experiments, we compared the quality of CF based on web usage mining with that of CF based on purchase data and then evaluated the effect of Clustering technique. Our experiments presented that the quality of CF with Web log data) better than CF with Purchase data. However, the application of clustering did not always give better performance.

5.2. Contributions

The research work presented in this paper makes the following contributions to the recommender systems related research community.

(1) Application of the k-means clustering algorithm to improve scalability of recommender systems.

(2) Development of a clickstream analysis technique to capture implicit ratings by tracking customer shopping behavior on the Web and its application to reduce the sparsity.

(3) Development of a methodology (clustering + CF based on Web usage mining) to apply data mining techniques for enhancing collaborative recommendations, in which Web usage mining and clustering algorithm is applied to address sparsity, scalability issues together.

(4) Suggestion of methodologies and evaluation of them with the real Internet shopping mall data to compare the effect of each approach.

5.3. Future Works

While our experimental results suggest that the proposed methodology are promising new recommendation methodology, these results are based on studies limited to the particular e-commerce site that has small customers, products, and transactions. Therefore, it is required to evaluate our methodologies in more detail using data sets from a variety of large e-commerce sets. As future works, it will be interesting to compare our suggested methodologies with one of

outstanding approaches to reduce the dimensionality of recommender system databases in the aspect of recommendation performance. And it will be also an interesting research area to conduct a real marketing campaign to customers using our methodologies and to evaluate their performance.

References

[1] Alsabti, K., Ranka, S., & Singh, V., "An Efficient K-Means Clustering Algorithm", *Proc. First Workshop on High-Performance Data Mining*, 1998.

[2] Bradley, P. S., Fayyad, U. M. and Reina, C., "Scaling Clustering Algorithms to Large Databases", *In Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD98)*, 1998, pp. 9-15.

[3] Breese, J. S., Heckerman, D., and Kadie, C., "Empirical analysis of predictive algorithms for collaborative filtering", *In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, 1998, pp. 43-52.

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

[5] Cooley, R., Mobasher, B., and Srivastava, J., "Data preparation for mining World Wide Web browsing patterns", *Journal of Knowledge and Information Systems*, 1, 1999.

[6] Goldberg, D., Nichols, D., Oki, B. M. & Terry, D., "Using Collaborative Filtering to Weave an Information Tapestry", *Communications of the ACM*, 35(12), 1992, pp. 61-70.

[7] Han, J. and Kamber, M., *Data mining: concepts and techniques*, Morgan Kaufmann Publishers, 2001.

[8] Hill, W., Stead, L., Rosenstein, M., and Furnas, G., "Recommending and evaluating choices in a virtual community of use", *In Proceedings of the 1995 ACM Conference on Human Factors in Computing Systems*, 1995, pp. 194-201.

[9] Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R. and Riedl, J., "GroupLens: applying collaborative filtering to usenet news", *Communications of the ACM*, 40, 3, 1997, pp. 77-87.

[10] Lee, J., Podlaseck, M., Schonberg, E. and Hoch, R., "Visualization and analysis of clickstream data of online stores for understanding web merchandising", *Data Mining and Knowledge Discovery*, 5, 1-2, 2001, pp. 59-84.

[11] Melville, P., Mooney, R.J, and Nagarajan, R., "Content-Boosted Collaborative Filtering", *In the Proceedings of the SIGIR-2001 Workshop on Recommender Systems*, 2001.

[12] Menascé, D.A., Almeida, V.A., Fonseca, R. and Mendes, M.A., "A methodology for workload characterization of e-commerce sites", *In Proceedings of ACM E-Commerce Conference*, 1999, pp. 119-128.

[13] Mobasher, B., Dai, H., Luo, T., Sun, Y. and Zhu, J., "Integrating Web usage and content mining for more effective personalization", *In Proceedings of the EC-Web* 2000, 2000, pp. 165-176.

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

[15] Rud, O., *Data mining Cookbook: Modeling Data for Marketing, Risk, and Customer Relationship Management*, Wiley & Sons Inc, 2001.

[16] Sarwar, B., Karypis, G., Konstan, J. and Riedl, J., "Item-based collaborative filtering recommendation algorithm", *In Proceedings of The Tenth International World Wide Web Conference*, 2001, pp. 285-295.

[17] Schafer, J. B., Konstan, J. A., Riedl, J., "E-commerce recommendation applications", *Data Mining and Knowledge Discovery*, 5 (1-2), 2001, pp. 115-153.

[18] Srivastava, J., Cooley, R., Deshpande, M. and Tan P., "Web usage mining: discovery and applications of usage patterns from web data", *SIGKDD Explorations*, 1, 2, 2000, pp. 1-12.

[19] Ungar, L. H., Foster, D. P., "A formal statistical approach to collaborative filtering", *In Proc. Conference on Automated Learning and Discovery*, 1998.

[20] VanderMeer, D., Dutta, K., Datta, A., "Enabling scalable online personalization on the Web", *In Proceedings of ACM E-Commerce Conference*, 2000, pp. 185-196.

요약

인터넷 쇼핑몰을 위한 데이터마이닝 기반 개인별 상품추천방법론의 개발

김재경* · 안도현* · 조윤호**

상품추천시스템은 고객들에게 추천 상품 리스트를 만들어 고객들이 구매 가능성이 있는 상품을 쉽게 찾도록 도와주는 개인화 된 정보필터링 기술이다. 협업 필터링(collaborative filtering)이 가장 성공적인 상품추천 기법으로 알려져 있으며 많이 이용되고 있다. 그러나, 인터넷 쇼핑몰에서 관리하는 상품과 고객의 수가 급속히 증가하면서 협업필터링에 기반한 상품추천 시스템은 입력 데이터의 희박성(Sparsity) 문제와 시스템 확장성(Scalability) 문제가 노출되고 있다. 따라서 본 연구에서는 협업필터링 기반 상품추천시스템의 상품추천 효과 및 성능을 개선하기 위해 웹 마이닝과 군집분석 기법에 기반을 둔 개인별 상품추천 방법론을 개발한다. 또한 실제 인터넷 쇼핑몰에서 개인별로 상품을 추천할 때 개발된 상품추천 방법론을 적용하여 다른 기존 상품추천 방법론과 실험적으로 비교함으로써 개발 방법론의 효과 및 성능을 검증한다.

* 경희대학교 경영대학 e비즈니스 전공
** 동양공업전문대학 인터넷정보과