Expert Systems with Applications 29 (2005) 700–704

Expert Systems with Applications

www.elsevier.com/locate/eswa

Classification-based collaborative filtering using market basket data

Jong-Seok Lee, Chi-Hyuck Jun*, Jaewook Lee, Sooyoung Kim

Department of Industrial Engineering, Pohang University of Science and Technology, San 31 Hyoja-dong, Pohang 790-784, South Korea

Abstract

Collaborative filtering based on voting scores has been known to be the most successful recommendation technique and has been used in a number of different applications. However, since voting scores are not easily available, similar techniques should be needed for the market basket data in the form of binary user-item matrix. We viewed this problem as a two-class classification problem and proposed a new recommendation scheme using binary logistic regression models applied to binary user-item data. We also suggested using principal components as predictor variables in these models. The proposed scheme was illustrated with a numerical experiment, where it was shown to outperform the existing one in terms of recommendation precision in a blind test.

© 2005 Elsevier Ltd. All rights reserved.

Keywords: Binary logistic regression; Classification; Collaborative filtering; Market basket data; Principal component analysis

1. Introduction

As a new marketing strategy recommender system plays an important role particularly in an e-commerce environment. It predicts the products which a customer is likely to select based on his or her past purchasing behavior and makes recommendations.

Although a variety of recommendation techniques has been developed recently, collaborative filtering (CF) has been known to be the most successful recommendation technique and has been used in a number of different applications such as recommending web pages, movies, articles and products (Hill, Stead, Rosenstein, & Furnas, 1995; Lee, Kim, & Rhee, 2001; Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994; Shardanand & Maes, 1995). The CF, first named by Goldberg, Nichols, Oki, and Terry (1992), assumes that a good way to find a certain customer's interesting content is to find other people who have similar interests with him. The goal of the CF is to suggest new items or to predict the utility of a certain item for a particular user based on the user's previous likings and the opinions of other like-minded users (Sarwar, Karypis, Konstan, & Riedl, 2001). Breese, Heckerman, and Kadie (1998) classify the CF into two classes: memory-based approach, also called as user-based approach, and model-based approach. The work by Goldberg et al. (1992) belongs to a user-based approach since it does not utilize any underlying model. There are some model-based CF methodologies, by using various tools such as Bayesian network, clustering, rule-based approach, Eigentaste and so on (Breese et al., 1998; Goldberg, Roeder, Gupta, & Perkins, 2001).

The CF is originally based on voting scores that customers express their preferences. However, customers seldom, if ever, vote on the products they used. It makes the CF suffer from the sparsity problem, one of its major problems, and companies endeavor to obtain more scores. To overcome such a problem, some researchers proposed a new CF scheme using market basket data (Mild & Reutterer, 2001, 2003), which can be transformed into a so-called binary user-item matrix having customers (users) and products (items) consisting of ones (purchases) and zeros (non-purchases). This scheme has the advantage that there is no need to gather voting scores and so the probability of using possibly distorted scores may be reduced. However, it usually resulted in a poor recommendation accuracy.

To improve the recommendation performance, in this paper, we propose a model-based CF scheme which utilizes binary logistic regression models as a classification tool. The basic idea of our proposed method is as follows: since our goal is to predict whether a customer's zero will stay or change into one in the near future for the market basket data, we view this problem as the classification problem

* Corresponding author. Tel.: +82 54 279 2197; fax: +82 54 279 2870. *E-mail address:* chjun@postech.ac.kr (C.-H. Jun).

0957-4174/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2005.04.037

and apply a classification technique to generate a recommendation scheme.

This paper is structured as follows: Section 2 describes the existing user-based CF schemes which are based on voting scores as well as binary user-item matrix. Section 3 focuses on the proposed methodology, which is based on binary logistic regression model using principal components as predictor variables. Section 4 provides a numerical example to test the performance of the proposed scheme. Finally, concluding remarks will be described in Section 5.

2. Collaborative filtering based on market basket data

Let us first introduce the existing CF which is based on customers voting scores. Suppose that there are n users and m items in the score database. Let v_{ij} be the score of item j evaluated by user u_i ($i=1, 2,\ldots, n; j=1, 2, \ldots, m$). If I_i is the set of items on which user u_i has recorded scores, then the mean score for user u_i is

$$\bar{v}_i = \frac{1}{|I_i|}\sum_{j\in I_i} v_{ij}. \tag{1}$$

In the user-based CF, the predicted score by the active user u_a for the item j, P_{aj}, can be estimated as follows (Breese et al., 1998)

$$P_{aj} = \bar{v}_a + \kappa_a \sum_{i=1}^{n} w(a,i)(v_{ij} - \bar{v}_i) \tag{2}$$

where $w(a,i)$ is the weight given by below, reflecting similarity between the active user u_a and each user u_i

$$w(a,i) = \frac{\sum_j (v_{aj}-\bar{v}_a)(v_{ij}-\bar{v}_i)}{\sqrt{\sum_j (v_{aj}-\bar{v}_a)^2 \sum_j (v_{ij}-\bar{v}_i)^2}} \tag{3}$$

and κ_a is a normalizing factor such that the absolute values of the weights sum to unity

$$\kappa_a = \frac{1}{\sum_{i=1}^{n} |w(a,i)|} \tag{4}$$

Note that in Eq. (3), all the summations are over the items j for which both users u_a and u_i have recorded scores simultaneously, that is, $j\in I_i \cap \mathrm{I}_a$. Note also that this weight utilizes Pearson correlation coefficient as the similarity measure.

For the data set, we consider the market basket data with binary information where voting scores by users are not available. The market basket data only tells us whether a user has purchased a particular item or not. The scores v_{ij} is, therefore, defined by

$$v_{ij} = \begin{cases} 1, & \text{when } u_i \text{ purchased item } j \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

Here, $V=(v_{ij})$ is called a binary user-item matrix. Notice that in our data descriptions, CF formula given by (2) is still valid although the use of the following modified version of it is sometimes desirable (Mild & Reutterer, 2003).

$$P_{aj} = \kappa_a \sum_{i=1}^{n} w(a,i) v_{ij} \tag{6}$$

3. Proposed classification-based CF scheme

Assuming that the binary user-item matrix in (5) is available, our goal is to predict the preference of the active user u_a for the item j that has not been purchased yet. That is, we need to determine whether the item j will be purchased or not. This is a two-class classification problem where the item belongs to class '1' if it will be purchased or class '0', otherwise. Hence, we may generate a classification rule by using the user-item matrix V in (5) as a learning sample where the j-th column will be a response variable and the other columns can be predictor variables. Then, we predict the class of the item under consideration for the active user by plugging observed predictor variables for the active user into the classification rule. Note that the active user is not a part of the learning sample. So, our classification model for predicting the class of item j can be represented in a general form as follows

$$v_j = f_j(v_1, \ldots, v_{j-1}, v_{j+1}, \ldots, v_m) + \varepsilon_j, \quad j = 1, \ldots, m. \tag{7}$$

where v_j is the j-th column of the binary user-item matrix, f_j is a suitable classification function for the item j to be estimated and ε_j is a random error term. The model in (7) will be called the j-th model. In this paper, we will consider binary logistic models for the classification.

Since the number of items under consideration in reality is normally very large, the model should involve a large number of predictor variables. Moreover, there may exist high correlation among items, which cause inefficiency in estimating the model. Therefore, we propose using the principal component analysis (PCA) to generate new variables before establishing a model.

3.1. Principal component analysis

PCA is a technique for forming new variables or principal components which are linear combinations of original variables. It should be noted that the new variables are uncorrelated themselves. If a substantial amount of total variance in the data is accounted for by a few principal components, then only these few new variables may be sufficient for the further analysis. Hence, PCA is commonly referred to as a dimension-reduction technique. PCA can be done by a spectral decomposition of the covariance matrix (Sharma, 1995).

Let $\boldsymbol{\Sigma}_j$ be the $(m-1)\times(m-1)$ covariance matrix of the binary user-item matrix excluding the j-th column. Then, its

Table 1
Contingency table for columns v_s and v_t

		v_t		
		1	0	Total
v_s	1	a	b	$a+b$
	0	c	d	$c+d$
	Total	$a+c$	$b+d$	n

s-th diagonal element is Var[v_s] and (s, t)-th element is Cov[v_s, v_t]. To estimate these elements, the contingency table shown in Table 1 can be utilized. Here, the cell statistics are defined as follows:

a the number of users that $v_s=1$ and $v_t=1$,
b the number of users that $v_s=1$ and $v_t=0$,
c the number of users that $v_s=0$ and $v_t=1$,
d the number of users that $v_s=0$ and $v_t=0$.

Now, the diagonal element of $\mathbf{\Sigma}_j$ can be estimated by

$$\mathrm{Var}[v_s] = \frac{a+b}{n}\frac{c+d}{n}. \tag{8}$$

Also, the off-diagonal element of $\mathbf{\Sigma}_j$ can be estimated by

$$\mathrm{Cov}[v_s,\ v_t] = \frac{a}{n} - \frac{a+b}{n}\frac{a+c}{n} \tag{9}$$

Next, we decompose the covariance matrix $\mathbf{\Sigma}_j$ spectrally into a product of two matrices, **P** and **Λ** such that

$$\mathbf{\Sigma}_j = \mathbf{P\Lambda P}' \tag{10}$$

where **Λ** is a diagonal matrix whose elements are the eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_{m-1}$ of the covariance matrix $\mathbf{\Sigma}_j$, and **P** is a $(m-1)\times(m-1)$ orthogonal matrix whose j-th column is the eigenvector corresponding to the j-th eigenvalue. Each element of the eigenvector becomes a weight to form a principal component. The number of principal components can be determined by observing the magnitude of eigenvalues.

3.2. Binary logistic regression model based on PCA

Suppose that the number of principal components is chosen as p. Then, these principal components for the j-th model can be formed as follows

$$\begin{aligned}\xi_1^j &= w_{11}^j v_1 + \cdots + w_{1j-1}^j v_{j-1} + w_{1j+1}^j v_{j+1} + \cdots + w_{1m}^j v_m \\ \xi_2^j &= w_{21}^j v_1 + \cdots + w_{2j-1}^j v_{j-1} + w_{2j+1}^j v_{j+1} + \cdots + w_{2m}^j v_m \\ &\vdots \\ \xi_p^j &= w_{p1}^j v_1 + \cdots + w_{pj-1}^j v_{j-1} + w_{pj+1}^j v_{j+1} + \cdots + w_{pm}^j v_m\end{aligned}, \tag{11}$$

where w_{kl}^j is the weight of l-th variable for the k-th new variable (principal component) used in the j-th model. So, the number of predictor variables used for the j-th model reduces from $m-1$ to p.

Utilizing these principal components, the j-th logistic regression model can be built as follows

$$P\{v_j = 1\} = P^j = \frac{\exp(\boldsymbol{\beta}^j \xi^j)}{1+\exp(\boldsymbol{\beta}^j \xi^j)}. \tag{12}$$

where ξ^j is a vector of principal components and $\boldsymbol{\beta}^j$ is the parameter vector to be estimated. Here, $\boldsymbol{\beta}^j$ can be estimated by the maximum likelihood estimation using an iterative technique such as Newton–Raphson method.

Once the estimate of $\boldsymbol{\beta}^j$ is obtained, the probability given in (12) can be estimated for the active user. To do this, purchase/non-purchase data of the active user for the rest of items should be used in evaluating ξ^j. The classification rule for the active user will be given by below

$$\begin{cases}\text{classify item } j \text{ into class 1 if } \hat{P}^j > 0.5 \\ \text{classify item } j \text{ into class 0 if } \hat{P}^j \leq 0.5\end{cases} \tag{13}$$

where $\hat{P}^j$ is the estimate of the probability in (12). If there are many active users, the above rule should be applied to each of the active users to estimate purchase probabilities.

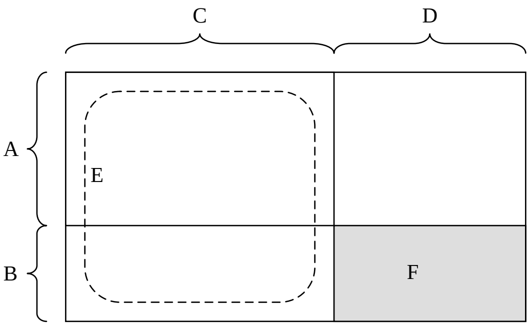
Similar procedure should be repeated in order to estimate purchase probabilities of other items under consideration. Then, we can recommend the N items corresponding to the N highest probabilities.

4. Numerical experiments

4.1. Experimental dataset

To evaluate the proposed method, the EachMovie data set, available on the web site, http://www.research.digital.com/SRC/eachmovie/, was used. This historical data set consists of 2,811,983 voting scores from 72,916 users on 1628 movies and videos. The voting scores are on a numeric six-point scale with [0.0, 0.2, 0.4, 0.6, 0.8, 1.0]. However, our proposed scheme requires a market basket data or a binary user-item matrix and so we need some modifications. Again, we are not arguing that we should always transform voting scores to binary data, but we would like to demonstrate that our model still works even when only binary data is available. For our purpose we changed the values of voted cells into ones and the null values of non-voted cells into zeros. For this experiment 725 users and 257 movies were randomly selected, which results that the proportion of ones for this subset is about 27.79%.

The data set prepared as above is divided into a training set (A) and a test set (B) as depicted in Fig. 1. The data for the training set is used as the CF database to build models. We then go through every user in the test set for making recommendations, treating each user as the active user. We also divide the items for active users into a set of items that treated as predictor variables (C) and a set that we will attempt to predict, in other words, a set consisting of items



A : Training users
B : Active users
C : Items referred to as independent variables
D : Items referred to as dependent variables
E : Data for calculating the similarities between A and B
in user-based approach
F : Comparison data set

Fig. 1. Division of the experimental data set.

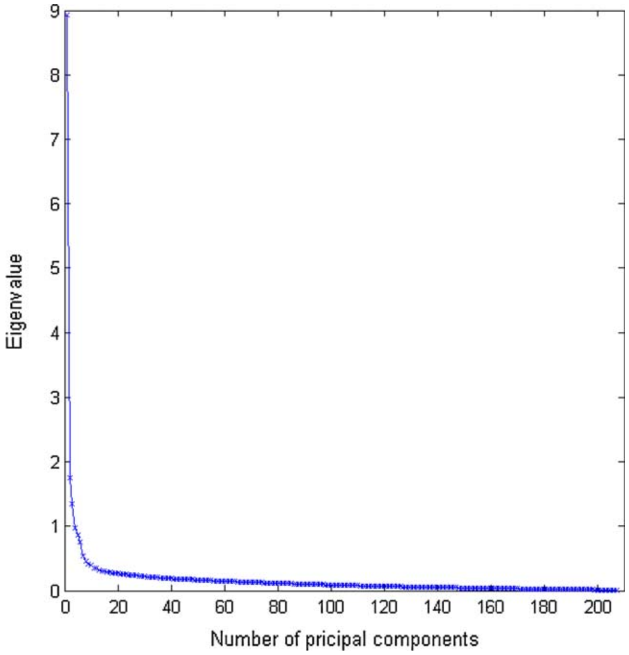


Fig. 2. Scree plot of eigenvalues.

regarded as response variables (D). Note that these divisions were done randomly. The information of the four sections in Fig. 1 is described in Table 2.

The area by dotted line (E) will be used to calculate similarities between training users and test users in the user-based approach by Eq. (6). The grey shaded area of the data (F) will be first blinded and then used to measure performance of the CF scheme. The division of the data set is illustrated in Fig. 1.

Our performance measure is the precision, which is generally used in information retrieval research and defined by

$$\text{Precision} = \frac{\text{hitting number}}{\text{Top} - N} \tag{14}$$

where ‘Top-N’ is the number of first N items that are recommended by a CF scheme and ‘hitting number’ is the actual Top-N obtained from the section F. We will consider various values of N ranging from 1 to 10.

Table 2
Summary of each section of data

Section	No. of users	No. of items	Non-zeros	Proportion of ones (%)
A×C	604	207	33,972	27.17
A×D	604	50	6519	21.59
B×C	121	207	9234	36.87
B×D	121	50	2046	33.82

4.2. Results

As mentioned in Section 3.1, we reduced the dimension of predictor variables by using PCA. The eigenvalues corresponding to the first 50 principal components are listed in Table 3. Fig. 2 shows all 207 eigenvalues as a scree plot, from which the first 10 principal components were chosen as predictor variables for our proposed scheme.

Using these 10 predictor variables, we built the 50 binary logistic regression models to estimate the preference of each of 121 active users for 50 items in the section F. The model parameters were estimated by the maximum likelihood estimation. Note that we need just one weight matrix for (11) to convert from the original binary variables to the principal component scores.

Table 3
Eigenvalue of each principal component

1	8.930	11	0.356	21	0.261	31	0.216	41	0.186
2	1.754	12	0.344	22	0.259	32	0.214	42	0.185
3	1.348	13	0.325	23	0.250	33	0.210	43	0.184
4	0.976	14	0.307	24	0.245	34	0.207	44	0.180
5	0.872	15	0.299	25	0.241	35	0.204	45	0.180
6	0.761	16	0.289	26	0.236	36	0.200	46	0.177
7	0.533	17	0.285	27	0.234	37	0.196	47	0.176
8	0.453	18	0.278	28	0.232	38	0.195	48	0.173
9	0.406	19	0.273	29	0.229	39	0.192	49	0.171
10	0.394	20	0.272	30	0.225	40	0.189	50	0.170

Table 4
Average precisions in percentage of two schemes

Top-*N*	Proposed	User-based
1	92.562	89.256
2	92.149	86.777
3	91.736	86.226
4	90.702	84.711
5	89.256	81.157
6	85.537	78.788
7	83.235	76.860
8	80.165	73.760
9	77.502	71.717
10	75.207	70.413

Table 4 shows the average precision of the proposed scheme over 121 users which is calculated by (14) according to various values of *N*. In this table the average precision of the existing user-based approach is also included for comparison. As we can see, the precision of the proposed approach is absolutely higher than the precision of the user-based approach. Fig. 3 compares first five values (that is, Top-1 to Top-5), which clearly shows the performance of the proposed scheme. The gap tends to be larger as *N* increases.

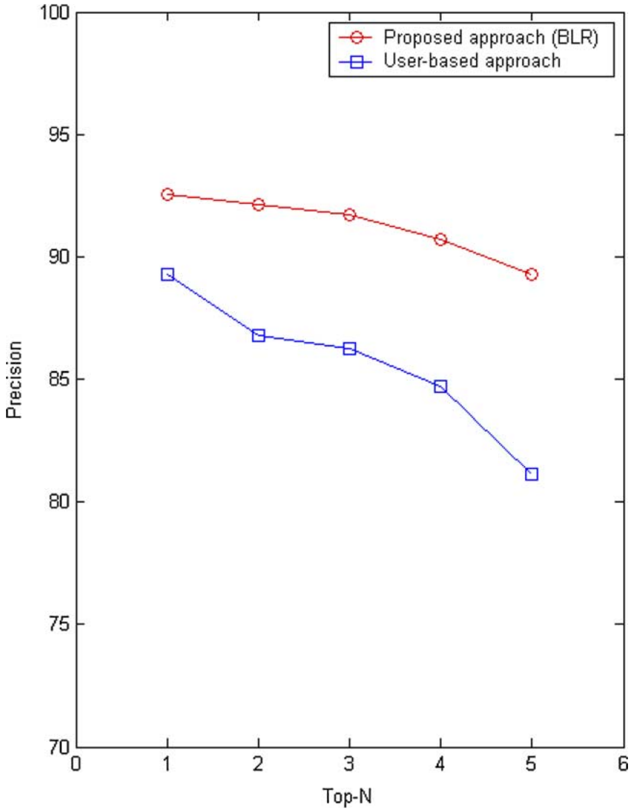


Fig. 3. Comparison of two schemes in average precision.

5. Conclusions

In this paper, we proposed a new approach to the model-based collaborative filtering (CF) under a binary user-item matrix based on the market basket. Binary logistic regression models combined with the principal component analysis were employed in the proposed CF scheme. Through the numerical experiments using a real data set, we evaluated the proposed scheme and demonstrated a significant performance improvement.

There may be a practical issue regarding computational time for on-line recommendation. The model training process in our proposed scheme can be made by off-line operations where the computational time of recommendation for an active user may be short enough. However, for the on-line recommendation system, a regular effort for updating the model should be necessary.

Acknowledgements

This work was supported by most (KOSEF) through National Core Research Center for Systems Bio-Dynamics.

References

Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering, *Microsoft research technical report*, MSR-TR-98-12.

Goldberg, D., Nichols, D., Oki, B., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, *35*(12), 61–70.

Goldberg, K., Roeder, T., Gupta, D., & Perkins, C. (2001). Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval Journal*, *4*(2), 133–151.

Hill, W., Stead, L., Rosenstein, M., & Furnas, G. (1995). *Recommending and evaluating choices in a virtual community of use Proceedings of the 1995 ACM conference on human factors in computing systems.* pp. 194–201.

Lee, C.-H., Kim, Y.-H., & Rhee, P.-K. (2001). Web personalization expert with combining collaborative filtering and association rule mining technique. *Expert Systems with Applications*, *21*(3), 131–137.

Mild, A., & Reutterer, T. (2001). Collaborative filtering methods for binary market basket data analysis. *Lecture Notes in Computer Science*, *2252*, 302–313.

Mild, A., & Reutterer, T. (2003). An improved collaborative filtering approach for predicting cross-category purchase based on binary market basket data. *Journal of Retailing and Consumer Services*, *10*, 123–133.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). *GroupLens: An open architecture for collaborative filtering of netnews Proceedings of the ACM 1994 conference on computer supported cooperative* pp. 175–186.

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th international world wide web conference (WWW10)*, Hong Kong, May 01–05. pp. 285–295.

Shardanand, U., & Maes, P. (1995). *Social information filtering: Algorithms for automating word of mouth Proceedings of ACM CHI'95 conference on human factors in computing systems.* pp. 210–217.

Sharma, S. (1995). *Applied multivariate techniques*. New York: Wiley (pp. 58–87).