

19th International Conference on Knowledge Based and Intelligent Information and Engineering Systems

Recommendation system for grocery store considering data sparsity

Natsuki Sano^{a,*}, Natsumi Machino^b, Katsutoshi Yada^c, Tomomichi Suzuki^a

^a*Tokyo University of Science, 2641 Yamazaki, Noda 278-8510, Japan*

^b*Recruit Communications Co., Ltd. 1-13-1 Kachidoki, Chuo-ward 104-0054, Japan*

^c*Kansai University, 3-3-35 Yamate-cho, Suita 564-8680, Japan*

Abstract

In grocery stores, large-scale transaction data with identification, such as point of sales (POS) data, is being accumulated as a result of the introduction of frequent shopper programs. We propose two recommendation systems based on transaction data of a grocery store. In recommending product items in grocery stores, data sparsity is a problem. This is because individual customers only purchase very few of the total number of product items a store sells. We evaluate various recommendation methods including SVD-type recommendation based on real POS data and summarize methods suitable for the proposed recommendation systems.

© 2015 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

Peer-review under responsibility of KES International

Keywords: Recommendation; grocery store; data sparsity; singular value decomposition; nonlinear principal component analysis

1. Introduction

In grocery stores, large-scale transaction data with identification, such as point of sales (POS) data, is being accumulated as a result of the introduction of frequent shopper programs (FSPs). The accumulated POS data have been used to examine customer shopping behavior, especially by professionals in the marketing field^{1, 2}.

Although the recommendations based on this data are often adopted in e-commerce shopping stores³, they are rarely introduced in face-to-face selling, such as in brick-and-mortar grocery stores. Therefore, introducing a system based on these recommendations to grocery stores could induce customers to visit the store to make a purchase.

We propose two recommended systems based on stored POS data, and these are shown in Figure 1. The first system gathers the e-mail address during the registration procedure and directly determines recommended products based on the stored POS data and sends reminders with discount information to customers by e-mail as shown in Figure 1 (a).

When constructing this system for grocery stores, the sparsity of evaluation values presents a problem. Evaluation values are constructed based on customers' purchase frequency of product items and is very sparse. This is because individual customers only purchase very few of the total number of product items a store sells.

We alleviated the problem of data sparsity by proposing a system in which recommended product items are determined by a two-step procedure as shown in Figure 1 (b). First, the system determines recommended product categories

* Corresponding author. Tel.: +81-4-7124-1501-3841 ; fax: +81-4-7122-4566.
E-mail address: nsano@rs.tus.ac.jp

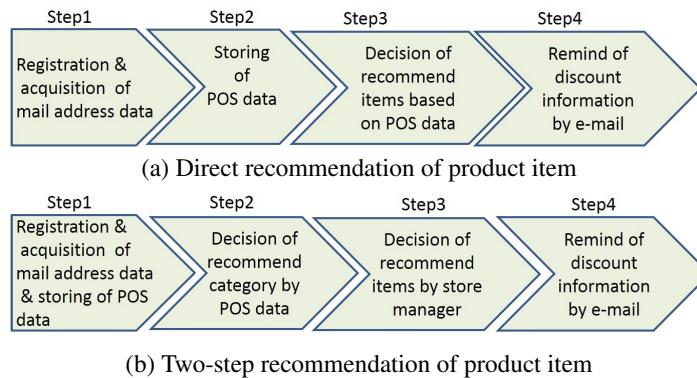


Fig. 1. Recommendation system for grocery stores

enabling the store manager to determine recommended product items according to the circumstances of the particular grocery store; for example, specific product items may have to be cleared out from inventory or could be purchased at a lower price than usual from the manufacturer. Discount information relating to the recommended product items are sent to remind customers by e-mail similar to the direct system.

The problem of data sparsity is often addressed technically by adopting singular value decomposition (SVD) in the recommended system^{3, 4, 5}. In our work we evaluate various recommended methods, including SVD-type recommendations, based on real POS data. A representative recommendation method is (i) user-based collaborative filtering (CF). A representative SVD-type recommendation method is (ii) recommendation based on evaluation values reconstructed by SVD, which factorizes the original evaluation matrix, thereby reducing it to low rank matrices and reconstructing the evaluation matrix. We also evaluate another type of SVD recommendation, (iii) recommendation by item-user similarity by SVD and (iv) recommendation by using a combination of CF and SVD. In addition, we evaluate (v) recommendation by nonlinear principal component analysis (NL-PCA)⁶, also known as sandglass-type neural networks (SNN).

The results of numerical experiments show that recommendation by SVD reconstruction is suitable for the recommendation of product items via the direct recommendation method, whereas CF is suitable for the recommendation of product categories via the two-step recommendation method.

The remainder of this article is organized as follows. Section 2 describes the POS data obtained from a real grocery store and the approach followed to construct a matrix of evaluation values. Section 3 explains the recommendation methods that were examined in the numerical experiment using real data. Section 4 presents the comparison results for product item and product category recommendations. Section 5 presents a summary of the study and offers conclusions and subjects for future studies.

2. Data Description

We employ POS data with identification gathered during May and June of 2009 in the Kanto area of Japan. The POS data reflect customers' purchase behavior, namely, who purchases what item, when, how many, and how much. In addition, the POS data contain information about the product category, that is, a large classification of the purchased product items. Examples of product items and categories are shown in Table 1.

We estimated the user's evaluation value of a product item (category) by using a scale from 0 to 5 by aggregating the number of purchases (i.e., the purchase frequency) of the specific product item (category) for each user. If the purchase frequency of items exceeded 5, then the evaluation value was considered as 5. The format of the matrix consisting of the evaluation values X is listed in Table 2.

Each row and column corresponds to the respective product item and user. An element of X , x_{ij} denotes the evaluation value of product item i by customer j . n and p denote the number of product items and users, respectively. When the evaluation value matrix for product item recommendation was constructed, the values that were used for n and p were 8674 and 6997, respectively.

Table 1. Examples of product items and categories

Product item	Product category
Takanashi low-fat milk (1L)	Milk
CGC 100% apple juice (1L)	Juice
Morinaga pudding (85g×3)	Dessert
Nippon ham margherita pizza (1 whole)	Pizza
Yamazaki double soft (6 slices)	Bread
Meiji bulgar plain yogurt (450g)	Yogurt
Yukijirushi butter (200g)	Butter & Margarine
Morinaga pino	Ice cream
QBB baby cheese (4 pieces 60g)	Cheese
Iodine egg hikari (6 eggs)	Egg
Shimadaya udon	Noodle
CGC soybean tofu kinu (350g)	Tofu
⋮	⋮

Table 2. Construction of evaluation matrix

(a) Matrix of evaluation value of product item (category)

	customer 1	customer 2	...	customer <i>j</i>	...	customer <i>p</i>
product item (product category) 1	x_{11}	x_{12}	...	x_{1j}	...	x_{1p}
product item (product category) 2	x_{21}	x_{22}	...	x_{2j}	...	x_{2p}
⋮	⋮	⋮	⋮	⋮	⋮	⋮
product item (product category) <i>i</i>	x_{i1}	x_{i2}	...	x_{ij}	...	x_{ip}
⋮	⋮	⋮	⋮	⋮	⋮	⋮
product item (product category) <i>n</i>	x_{n1}	x_{n2}	...	x_{nj}	...	x_{np}

An evaluation value for product item recommendation is very sparse, since the number of product items a customer purchases is very few among all the products on offer *n*. This sparsity of evaluation values could result in product item recommendation performing poorly. We alleviated the sparsity problem by constructing a matrix of evaluation values for the product categories. For this evaluation the values of *n* and *p* were 228 and 6997, respectively.

The sparsity of the evaluation value for product items was compared with that for product categories by defining a measure of non-sparsity as the ratio of the number of non-zero elements in the matrix of evaluation values among all the elements, $n \times p$. The results of the determination of the sparsity measure are shown in Table 3.

Table 3. Comparison between sparsity of evaluation values of product items and categories

Product item	Product category
0.032%	8.75%

These results show that although the non-sparsity of product items has the value of 0.032, that of the product categories has the value of 8.75; therefore, the non-sparsity of the evaluation value matrix of product categories is increased 273.44 times compared to that of product items.

3. Recommendation methods for comparison

In this section we present a comparison between the five recommended methods. The predictive recommendation values for each method is computed for item *i* by user *j*, r_{ij} , $i = 1, \dots, n$, $j = 1, \dots, p$. For an active user *a*, the values of r_{ia} $i = 1, \dots, n$ are sorted in decreasing order and the top *m* and unpurchased items are recommended to active user *a*.

3.1. User-based collaborative filtering

Collaborative filtering is the most popular recommendation method and we employ user-based collaborative filtering, which computes a recommendation value based on similarity between users as follows:

$$r_{ij} = \bar{x}_j + \frac{\sum_{l=1}^N s_{lj} x_{lj}^*}{\sum_{l=1}^N |s_{lj}|}, \quad (1)$$

where \bar{x}_j denotes j -th user's mean $\bar{x}_j = (\sum_{l=1}^n x_{lj})/n$, $x_{ij}^* = x_{ij} - \bar{x}_j$ denotes user j 's evaluation of item i that subtracts user j 's mean \bar{x}_j , and s_{lj} denotes the similarity between user l and user j by using the Pearson correlation

$$s_{lj} = \frac{(\mathbf{x}_l - \bar{\mathbf{x}}_l)'(\mathbf{x}_j - \bar{\mathbf{x}}_j)}{\|\mathbf{x}_l - \bar{\mathbf{x}}_l\| \cdot \|\mathbf{x}_j - \bar{\mathbf{x}}_j\|}, \quad (2)$$

where $\mathbf{x}_j = (x_{1j}, x_{2j}, \dots, x_{nj})$ denotes j -th users evaluation vector and $\bar{\mathbf{x}}_j = (\bar{x}_j, \bar{x}_j, \dots, \bar{x}_j)$ denotes n -length replicates of j -th user's mean. Note that N denotes the number of neighborhoods and significant to the recommendation performance.

3.2. Recommendation by SVD-reconstructed evaluation values

$\hat{\mathbf{X}}_k$, the least-square approximation of \mathbf{X} , is obtained by using the SVD technique.

$$\hat{\mathbf{X}}_k = \mathbf{U}_k \mathbf{D}_k \mathbf{V}_k', \quad (3)$$

where \mathbf{U}_k and \mathbf{V}_k denotes $n \times k$ and $p \times k$ orthogonal matrix, \mathbf{D}_k denotes $k \times k$ diagonl matrix. \hat{x}_{ij} , the i - j element of $\hat{\mathbf{X}}$ is available for the predictive recommendation value of the i -th item for the j -th user, r_{ij} . The dimension of the latent class, k is a significant parameter for evaluating recommendation performance.

3.3. Recommendation by item-user similarity by SVD

Similarity between the i -th item and the j -th user by using SVD is expressed as follows:

$$s_{ij}^{user,item} = \frac{(\mathbf{u}_i - \bar{\mathbf{u}}_i)'(\mathbf{v}_j - \bar{\mathbf{v}}_j)}{\|\mathbf{u}_i - \bar{\mathbf{u}}_i\| \cdot \|\mathbf{v}_j - \bar{\mathbf{v}}_j\|}, \quad (4)$$

$i = 1, \dots, n, j = 1, \dots, p$, where \mathbf{u}_i denotes the i -th vector of \mathbf{U}_k and \mathbf{v}_j denotes the j -th vector of \mathbf{V}_k . Further, $\bar{\mathbf{u}}_i = (\bar{u}_i, \bar{u}_i, \dots, \bar{u}_i)$ denotes replicates of the mean of the i -th item $\bar{u}_i = (\sum_{l=1}^k u_{il})/k$ and $\bar{\mathbf{v}}_j = (\bar{v}_j, \bar{v}_j, \dots, \bar{v}_j)$ denotes replicates of the j -th user's mean $\bar{v}_j = (\sum_{l=1}^k v_{jl})/k$. $s_{ij}^{user,item}$ is also available for the predictive recommendation value of the i -th item of the j -th user, r_{ij} .

3.4. Recommendation by combination of CF and SVD

Although ordinal user-based collaborative filtering recommends prospective items based on similarities between users in Equation (2), we combine collaborative filtering and latent similarity between users by using SVD, such as in

$$s_{ij}^{user,user} = \frac{(\mathbf{v}_i - \bar{\mathbf{v}}_i)'(\mathbf{v}_j - \bar{\mathbf{v}}_j)}{\|\mathbf{v}_i - \bar{\mathbf{v}}_i\| \cdot \|\mathbf{v}_j - \bar{\mathbf{v}}_j\|}. \quad (5)$$

Equation (5) is substituted to s_{lj} in equation (1) and the predictive recommendation value of the i -th item for the j -th user, r_{ij} is calculated.

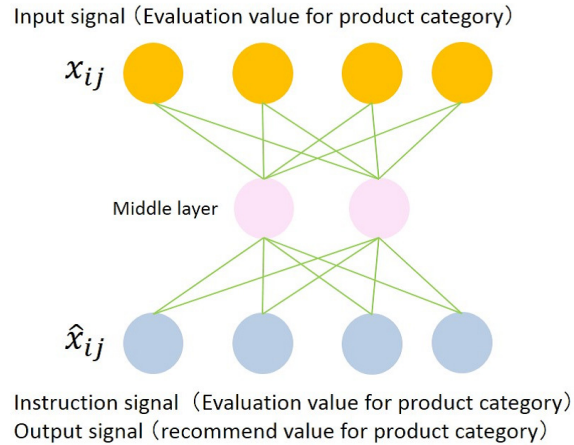


Fig. 2. Topology of sandglass-type neural networks

3.5. Recommendation by non-linear PCA

We apply non-linear PCA⁶ (NL-PCA) to recommendation systems by using sandglass-type neural networks (SNN). Vozalis⁷ applied non-linear PCA to recommendation by combining this method with collaborative filtering. We used the output values of SNN directly as predictive recommendation value. A topology of SNN is shown in Figure 2.

The input signal in the input layer and instruction signal in the output layer are the same and the input-output relation is learned through the back-propagation algorithm in NL-PCA. The number of units in the middle layer, u is smaller than the number of units in the input and output layer. Information from the input layer is compressed to reduce the dimension of the middle layer and reconstructed in the output layer. This means that NL-PCA uses a nonlinear reconstruction to execute the input signal.

In the proposed recommendation system, the input and instruction signals are evaluation values of the product items or categories and the number of units in the input and output layer is n . The weights between units are learned, such that the sum of the square error in the output layer is minimized. After learning, the evaluation values are once again input into the input units, and the output values in the output units are available for the predictive recommendation value of the i -th item for the j -th user, r_{ij} .

4. Numerical experiments based on real data

4.1. Performance measure

We sampled 2% of the elements from the evaluation matrix X , making up missing values intentionally, and complementing them with 0 values. In other words, we assumed that actually purchased products are not purchased and performed estimations by using the recommendation methods in Section 3. The sets consisting of the sampled items, hit items determined by the recommendation method, and the recommended items, were termed *testset*, *hitset*, and *recset* respectively. By using these sets, the recommendation methods are compared through precision, recall, and F-value as follows:

$$\begin{aligned} \text{recall} &= \frac{|hitset|}{|testset|} \\ \text{precision} &= \frac{|hitset|}{|recset|} \\ \text{F-value} &= \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}} \end{aligned}$$

We conducted the above procedures 10 times and evaluated the recommendation methods by using the average value of recall, precision, and F-value.

Table 4. Performance comparison of recommendation method for product items with $m = 5$

Recommendation method	optimal parameter	precision	recall	F-value
User-based CF	$N = 30$	0.0030	0.0271	0.0054
SVD reconstruction	$k = 17$	0.0040	0.0358	0.0072
SVD similarity	$k = 4$	0.0033	0.0296	0.0059
CF & SVD	$k = 17, N = 390$	0.0030	0.0273	0.0055
NL-PCA	–	–	–	–

4.2. Performance evaluation of item recommendation

We evaluated the recommendation algorithms for product item recommendation in terms of recall, precision, and F-value with the number of recommended items for active users, $m = 5$. The parameters of the recommendation methods were optimized by maximizing the F-value.

Table 4 shows the results of the optimized parameters and the value of the performance measure. The F-value of recommendation by SVD reconstruction shows the highest value (0.0072), followed by recommendation by SVD similarity (0.0059), recommendation by CF & SVD (0.0055), and CF (0.0054). The horizontal bar in the row of NL-PCA shows that the algorithm could not be executed, as the number of input units, i.e., the number of product items, n is 8574, which exceeds the admissibility condition of the number of input units in the `nnet` package of R.

4.3. Performance evaluation of category recommendation

For recommendation of the product category, we conducted an evaluation for each recommendation method. Table 5(a) lists the results of the optimized parameters and value of the performance measure with $m = 3$. The F-value of user-based CF shows the highest value (0.0429), followed by recommendation by SVD reconstruction (0.0407), recommendation by CF & SVD (0.0388), NL-PCA (0.0387), and SVD similarity (0.0036).

Table 5(b) lists the results of optimized parameters and values of the performance measure with $m = 4$. F-value of recommend by CF shows the highest value (0.0398), followed by recommend by SVD reconstruction (0.0375), recommend by CF & SVD (0.0363), NL-PCA (0.0357), and SVD similarity (0.0039).

Table 5(c) lists the results of the optimized parameters and values of the performance measure with $m = 5$. The F-value of recommendation by CF shows the highest value (0.0377), followed by recommendation by SVD reconstruction (0.0362), NL-PCA (0.0348), recommendation by CF & SVD (0.0347), and SVD similarity (0.0037).

Therefore, user-based CF is superior to the other recommendation methods for product categories for all values of m in all the cases. On the contrary, the recommend by SVD similarity is inferior to the other methods.

4.4. Performance comparison of item recommendation with category recommendation

We compared the best performance of product item recommendation with that of product category recommendation with $m = 5$. The F-value of recommendation by SVD reconstruction for product item recommendation is 0.0072. On the other hand, the F-value of recommendation by user-based collaborative filtering is 0.0377. We can observe that the F-value of the best recommendation method for product category recommendation is increased 5.24 times compared to the product item method.

5. Conclusion

When recommendation systems are applied to grocery stores, the sparsity of evaluation values can be a problem. This paper proposes two recommendation systems (a) direct recommendation and (b) two-step recommendation of product items based on stored POS data considering the sparsity of data. Two-step recommendation is composed of product category recommendation by using a recommendation algorithm and product item recommendation by heuristic decision of store manager. To seek the appropriate recommendation algorithm for the proposed algorithm, five

Table 5. Performance comparison of recommend method for product category

(a) $m = 3$

Recommend method	optimal parameter	precision	recall	F-value
User-based CF	$N = 220$	0.0245	0.176	0.0429
SVD reconstruction	$k = 4$	0.0233	0.166	0.0407
SVD similarity	$k = 6$	0.0021	0.0147	0.0036
CF & SVD	$k = 4, N = 470$	0.0222	0.159	0.0388
NL-PCA	$u = 2$	0.0219	0.165	0.0387

(b) $m = 4$

Recommend method	optimal parameter	precision	recall	F-value
User-based CF	$N = 90$	0.0220	0.210	0.0398
Approximation by SVD	$k = 2$	0.0208	0.198	0.0375
Similarity by SVD	$k = 6$	0.0021	0.0204	0.0039
CF & SVD	$k = 4, N = 420$	0.0202	0.192	0.0363
NL-PCA	$u = 3$	0.0196	0.197	0.0357

(c) $m = 5$

Recommend method	optimal parameter	precision	recall	F-value
User-based CF	$N = 280$	0.0205	0.245	0.0377
Approximation by SVD	$k = 2$	0.0197	0.234	0.0362
Similarity by SVD	$k = 6$	0.00201	0.0241	0.0037
CF & SVD	$k = 4, N = 320$	0.0189	0.225	0.0347
NL-PCA	$u = 2$	0.0179	0.225	0.0348

recommendation algorithms are compared based on real POS data. The results show that recommendation according to the reconstructed evaluation value by SVD is appropriate for direct recommendation and user-based collaborative filtering is appropriate for two step recommendation. Decision of specific procedure of product item recommend by store manager in two-step recommend is a subject for future study.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 25750131 and the Strategic Project to Support the Formation of Research Bases at Private Universities: Matching Fund Subsidy from MEXT (Ministry of Education, Culture, Sports, Science and Technology).

References

1. Guadagni, P.M., Little, J.D.C.. A logit model of brand choice, calibrated on scanner data. *Marketing Science* 1983;2:203–238.
2. Sano, N., Yada, K.. The influence of sales areas and bargain sales on customer behavior in a grocery store. *Neural Computing and Applications* 2015;26(2):355–361.
3. Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J.T.. Application of dimensionality reduction in recommender system –a case study. In: *Proceedings of the ACM WebKDD Workshop*. 2000, .
4. Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J.T.. Analysis of recommendation algorithms for ecommerce. In: *Proceedings of the 2nd ACM conference on Electronic commerce*. 2000, p. 158–167.
5. Goldberg, K., Roeder, T., Gupta, D., Perkins, C.. Eigentaste : A constant time collaborative filtering algorithm. *Information Retrieval Journal* 2001;4:133–151.
6. Scholz, M., Fraunholz, M., , Selbig, J.. Nonlinear principal component analysis: Neural network models and applications. In: *Principal Manifolds for Data Visualization and Dimension Reduction*; vol. 58 of *Lecture Notes in Computational Science and enginee*. Heidelberg: Springer; 2008, p. 44–67.
7. Vozalis, M.G., Markos, A., Margaritis, K.G.. Collaborative filtering through svd-based and hierarchical nonlinear pca. In: *Proceedings of 2010 International Conference on Artificial Neural Networks*. 2010, p. 395–400.