Expert Systems with Applications 28 (2005) 823–830

Expert Systems with Applications

www.elsevier.com/locate/eswa

Collaborative filtering based on iterative principal component analysis

Dohyun Kim, Bong-Jin Yum*

Department of Industrial Engineering, Korea Advanced Institute of Science and Technology, 373-1 Gusung-Dong, Yusung-Gu, Taejon 305-701, South Korea

Abstract

Collaborative filtering (CF) is one of the most popular recommender system technologies, and utilizes the known preferences of a group of users to predict the unknown preference of a new user. However, the existing CF techniques has the drawback that it requires the entire existing data be maintained and analyzed repeatedly whenever new user ratings are added. To avoid such a problem, *Eigentaste*, a CF approach based on the principal component analysis (PCA), has been proposed. However, *Eigentaste* requires that each user rate every item in the so called gauge set for executing PCA, which may not be always feasible in practice. Developed in this article is an iterative PCA approach in which no gauge set is required, and singular value decomposition is employed for estimating missing ratings and dimensionality reduction. Principal component values for users in reduced dimension are used for clustering users. Then, the proposed approach is compared to *Eigentaste* in terms of the mean absolute error of prediction using the Jester, MovieLens, and EachMovie data sets. Experimental results show that the proposed approach, even without a gauge set, performs slightly better than *Eigentaste* regardless of the data set and clustering method employed, implying that it can be used as a useful alternative when defining a gauge set is neither possible nor practical.

Keywords: Recommender system; Collaborative filtering; Principal component analysis; Singular value decomposition

1. Introduction

Collaborative filtering (CF) is a frequently used technique in recommender systems. It is based on the assumption that if two users rate certain items similarly, they are likely to have similar tastes and will rate other items in a similar manner (Goldberg, Roeder, Gupta, & Perkins, 2001; Resnick, Iacocou, Suchak, Berstrom, & Riedl, 1994). CF has the advantage of providing the support for filtering such items as movies, pictures, etc. that are hard to analyze using an automated process. However, the existing CF techniques are not without a drawback. That is, they require the entire existing data to be maintained and analyzed repeatedly whenever new user ratings are added. To avoid this problem, the principal component analysis (PCA) has been introduced to CF (Goldberg et al., 2001) to reduce the dimensionality of the data for computational efficiency.

PCA is a multivariate analysis technique that transforms a set of g correlated variables into a set of orthogonal components. Although g components are required to reproduce the total variability of the original variables, much of this variability can be often explained by a small number k of the principal components (PC's). The k PC's can then replace the g original variables, thus compressing the original data set into a smaller one that consists of the k PC's (Johnson & Wichern, 2002).

Since a data set of user ratings usually contains many missing values, it is first necessary to devise a method for dealing with missing values before conducting PCA. For instance, *Eigentaste*, a PCA-based CF algorithm developed by Goldberg et al. (2001), defines a gauge set that consists of those items rated by all users. The gauge set is then used to execute PCA. However, it may not be always feasible to obtain such a gauge set since users may have different experiences and backgrounds. In addition, if PCA is conducted only on the items in a specific gauge set, the information contained in the remaining user ratings will be lost. As an alternative, case deletion or imputation is often used to obtain a complete data matrix (Adams, Walczak, Vervaet, Risha, & Massart, 2002). Case deletion omits all variables with any missing value in the data matrix. Since very few users or items could be left after its execution, case deletion is not appropriate for CF. On the other hand, in the imputation method, a missing value is filled with

* Corresponding author. Tel.: +82 42 869 3116; fax: +82 42 869 3110.
E-mail address: bjyum@kaist.ac.kr (B.-J. Yum).

0957-4174/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2004.12.037

a plausible one. For example, a column or a row average of the data matrix may be substituted for a missing value. Nonetheless, such imputation is known to have the risk of distorting the covariance structure of the data (Adams et al., 2002).

In this article, an iterative PCA-based method is developed for CF without defining a gauge set. The proposed iterative PCA-based method simultaneously estimates the missing values and determines the PC's using singular value decomposition (SVD). Then, PC values of users in reduced dimension are used for clustering users. The proposed approach and *Eigentaste*, combined with two clustering methods, are compared in terms of the mean absolute error (MAE) of prediction using three data sets (i.e. Jester (Goldberg et al., 2001), MovieLens (MovieLens and GroupLens websites), and EachMovie (EachMovie website) data sets).

It is worth noting that SVD has been used in CF for estimating missing values and dimensionality reduction. For instance, Sarwar, Karypis, Konstan, and Riedl (2000) compared relative performances of an SVD-based and traditional CF approaches using explicit ratings as well as product purchase (i.e. binary) data, and found that the SVD-based approach yields better results unless the user-item matrix is extremely sparse. The proposed approach is similar to Sarwar et al. (2000) in that it deals with the case of explicit ratings and SVD-based. However, in Sarwar et al. (2000), SVD is performed just once on the imputed user-item matrix and recommendations are made without clustering users for the case of explicit ratings, while, in the proposed approach, SVD is iteratively performed until convergence and users are clustered based on their PC values in reduced dimension.

2. *Eigentaste* algorithm (Goldberg et al., 2001)

In *Eigentaste*, the ratings of g gauge items are collected from all users to form a gauge set. Let $\mathbf{A}_g^*$ be the $m\times g$ matrix of ratings by m users on g gauge items. Each rating a_{ij}^* in $\mathbf{A}_g^*$ is normalized by subtracting the mean rating of the j-th item from a_{ij}^* and then dividing the result by the standard deviation of $\{a_{ij}^*,\ i=1,2,\ldots,m\}$. Let a_{ij} be the normalized rating and $\mathbf{A}_g(=[a_{ij}])$ be the normalized gauge matrix.

PCA of $\mathbf{A}_g$ is then conducted to obtain k PC's, which explain most of the variability of $\mathbf{A}_g\cdot k$ PC values of each user are then substituted for the ratings on g gauge items. In *Eigentaste*, k is set to two for visualization, although it easily generalizes to higher k's.

The two PC values for each user are plotted in two dimensions. Goldberg et al. (2001) observed that the plotted points for the Jester data show a high concentration around the origin, and used the recursive rectangular clustering (RRC) method for clustering users in the gauge set. The RRC method first constructs the minimal rectangular cell that encloses all the points in the plot, and then successively bisects the cells towards the origin until a desired depth is reached. The resulting cells are treated as the clusters of users. For a detailed description of the RRC method, the reader is referred to Goldberg et al. (2001).

For each cluster, the mean rating of each non-gauge item is calculated, and non-gauge items are sorted according to their mean ratings to form a lookup table. When a new user arrives, he/she is guided to rate all the items in the gauge set. The obtained ratings are then converted into two PC values, based on which the appropriate cluster for the new user is determined. Finally, appropriate recommendations from the lookup table are presented to the new user, and a rating for each recommendation is collected.

3. Proposed method

The proposed CF method consists of: (1) estimation of missing ratings in the existing user-item matrix and dimensionality reduction by the iterative PCA; (2) clustering of existing users using their PC values; and (3) making recommendations for a new users. In the following, these steps are explained in detail.

3.1. Iterative PCA

The iterative PCA method does not require that a gauge set be defined. Instead, it can be applied to the whole data set, iteratively executing PCA for estimating the missing values and identifying PC's simultaneously. The iterative PCA method adopted in the present study follows the algorithm of Walczak and Massart (2001) or Adams et al. (2002), and proceeds as described below (see also Fig. 1).

Let $\mathbf{A}^*$ be the $m\times n$ user-item matrix under consideration in which a_{ij}^* is the rating for item j by user i. If a_{ij}^* is missing, then it is initially filled with a value by some chosen method. In the present study, it is filled with the average of the corresponding column and row averages. In addition, let $\mathbf{A}_c$ be the centered matrix of $\mathbf{A}^*$ obtained by subtracting the mean rating of the j-th item from a_{ij}^* for $i=1,2,\ldots,m$ and $j=1,2,\ldots,n$. Then, in the course of iterations, SVD of $\mathbf{A}_c$ is conducted. The SVD of any $m\times n$ matrix $\mathbf{A}_c$ is defined as:

$$\mathbf{A}_c = \mathbf{U}\Sigma\mathbf{V}^{\mathbf{T}} \tag{1}$$

where '$\mathbf{T}$' denotes a transposition, $\mathbf{U}$ is the $m\times m$ matrix of the eigenvectors of $\mathbf{A}_c\mathbf{A}_c^{\mathbf{T}}$, $\mathbf{V}$ is the $n\times n$ matrix of the eigenvectors of $\mathbf{A}_c^{\mathbf{T}}\mathbf{A}_c$, and $\boldsymbol{\Sigma}$ is an $m\times n$ matrix in which σ_{ii} is a singular value of $\mathbf{A}_c$ for $i=1, 2, \ldots, \min(m, n)$ and the other elements are 0 (Golub & Van Loan, 1996). The SVD of any matrix has the property that, if a $k\times k$ diagonal matrix $\boldsymbol{\Sigma}_k$ is constructed by choosing the k ($<\min(m,n)$) largest elements among σ_{ii}'s, then the reconstructed matrix $\mathrm{A}_c'=\mathbf{U}_k\boldsymbol{\Sigma}_k\mathbf{V}_k^{\mathbf{T}}$ provides the best lower rank approximation of the matrix $\mathbf{A}_c$ where $\mathbf{U}_k$ is $m\times k$ and $\mathbf{V}_k$ is $n\times k$ (Golub & Van Loan, 1996). The missing values in $\mathbf{A}^*$ can be replaced with the corresponding values in $\mathbf{A}'$, the matrix obtained by performing

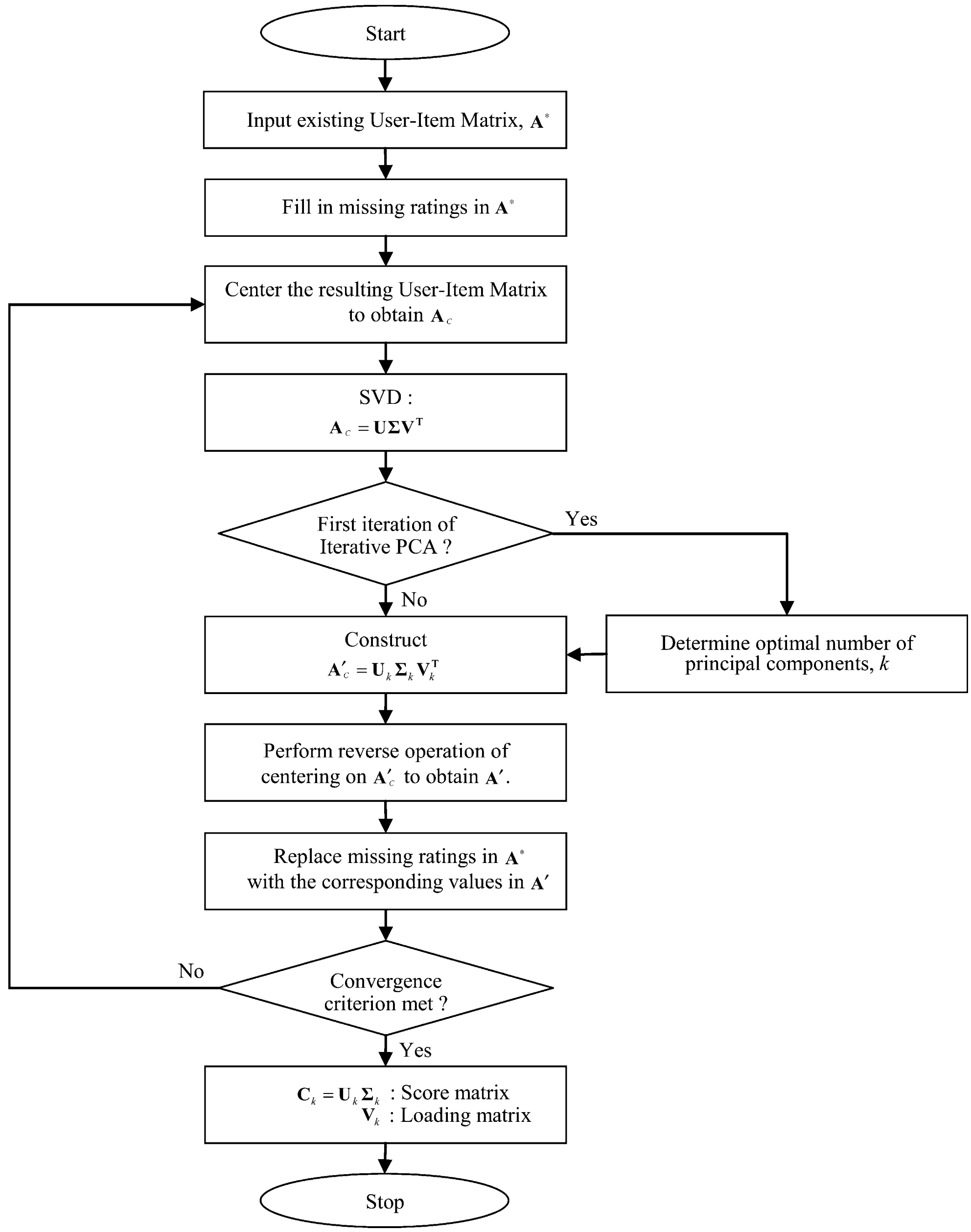

Fig. 1. Procedures for the iterative PCA.

a reverse operation of centering on $\mathbf{A}'_c$. The above process is repeated until some convergence criterion is met. In addition, $\mathbf{C}_k(=\mathbf{U}_k\boldsymbol{\Sigma}_k)$ corresponds to $m\times k$ matrix of k PC values (i.e. scores) of $\mathbf{A}_c$, and $\mathbf{V}_k$ represents the loading matrix for k PC's (Adams et al., 2002). Therefore, the proposed method estimates the missing values and finds the PC's simultaneously. The estimated missing ratings of the existing users are retained in the user-item matrix and are used to predict the missing ratings of new users later.

A scree plot (Johnson & Wichern, 2002) is used to determine the number of PC's (i.e. k) in the first iteration of the iterative PCA. That is, σ_{ii}^2 is plotted against $i=1,2,\ldots,\min(m,n)$ and k is set to i beyond which σ^2's are comparatively small and are of approximately the same size. Note that σ_{ii}^2's are equivalent to the eigenvalues of $\mathbf{A}_c^T\mathbf{A}_c$ and are proportional to the variances of the PC's of $\mathbf{A}_c$. In addition, the following convergence criterion is adopted, which is a modified version of the criterion in Adams et al. (2002):

$$\delta = \frac{|\mathrm{SS}_l - \mathrm{SS}_{l-1}|}{\mathrm{SS}_{l-1}} < c \tag{2}$$

where, l is the iteration number, $\mathrm{SS}_l = \sum_{q=1}^{Q}(\hat{h}_q)^2$, Q is the total number of missing values, $\hat{h}$ is the estimate of a missing value, and c is a small constant (10^{-10} in the present study).

3.2. Clustering

The principal component values obtained from the iterative PCA are used for clustering users. For the purpose of clustering, the proposed method employs the RRC method as well as the K-means clustering algorithm (MacQueen, 1967) for comparison.

The RRC method is described in Section 2. For K-means clustering, the Matlab *'kmeans'* function (Matlab Statistics Toolbox) was used. *kmeans* uses an iterative algorithm that minimizes the sum of distances from each object to its cluster centroid, over all clusters. It moves objects between clusters until the sum cannot be decreased further (Matlab Statistics Toolbox). For a detailed description, the reader is referred to Matlab Statistics Toolbox. The number of clusters (i.e. K) is determined based on the sum of squared errors criterion (Duda, Hart, & Stork, 2001). Let $\mathbf{p}_{iu}$ and $\boldsymbol{\mu}_i$ represent the column vectors of user u's PC values in cluster i and the centroid of cluster i, respectively. Then, the criterion is defined as

$$\mathrm{SS}_K = \sum_{i=1}^{K} \mathrm{SS}_i, \quad K = 2, 3, \ldots$$

where

$$\mathrm{SS}_i = \sum_{u \in C_i} (\mathbf{p}_{iu} - \boldsymbol{\mu}_i)^{\mathrm{T}} (\mathbf{p}_{iu} - \boldsymbol{\mu}_i), \quad i = 1, 2, \ldots, K$$

and C_i denotes the i-th cluster. In general, SS_K becomes the largest when K equals 2 and decreases as K increases. In the present study, we choose the smallest K for which $|\mathrm{SS}_K - \mathrm{SS}_{K+1}|/\mathrm{SS}_2$ is less than or equal to 0.001.

For the RRC method, the number of clusters (i.e. rectangular cells) may take values of 4, 16, 28, and so on in increments of 12, and K is selected among these values according to the above rule.

3.3. Prediction

The proposed method does not require a new user to rate the items in a gauge set, but guides a new user to rate items he/she chooses (recommendations can be made for a new user who rates at least one item). Then, the iterative PCA is executed using the available ratings from the new user and a portion of the existing user-item matrix to estimate his/her missing ratings and find PC values. After the iterative PCA, the cluster to which the new user belongs is determined and recommendations are made to the new user. Detailed procedures are as follows.

1. Select the latest $(m'-1)$ rows from the existing user-item matrix to form a $(m'-1)\times n$ sub-matrix.
2. Append the rating vector of a new user to the sub-matrix in Step 1 as the last row to form $m'\times n$ matrix $\mathbf{B}^*$.
3. The iterative PCA described in Section 3.1 is applied to $\mathbf{B}^*$ to estimate the missing ratings in the last row of $\mathbf{B}^*$. The number of PC's (i.e. k) is the same as the one determined in Section 3.1 for the existing user-item matrix. Let $\mathbf{b}^*$ be the resulting last row of $\mathbf{B}^*$.
4. Calculate the PC values of the new user by multiplying $\mathbf{b}^*$ by $\mathbf{V}_k$ where $\mathbf{V}_k$ is the loading matrix found in Section 3.1 using the existing user-item matrix.
5. From the clusters defined in Section 3.1 for the existing users, select the cluster to which the new user belongs.
6. Each missing rating of the new user is replaced with the average of the corresponding actual ratings by the users in the cluster, and recommendations are made to the new user.

When the RRC method is used, the cluster for the new user in Step 5 is determined by comparing its PC values to the coordinate values of each existing cluster (i.e. cell). On the other hand, when the K-means clustering method is used, the cluster whose centroid is closest to the PC values is selected for the new user.

4. Experimental results

Computational experiments were conducted using three data sets to compare *Eigentaste* and the proposed method in terms of prediction accuracy. In addition, two clustering methods, RRC and K-means algorithm, were compared in the experiment.

4.1. Data sets

The data sets used for the computational experiments include Jester (Goldberg et al., 2001), MovieLens (MovieLens and GroupLens websites), and EachMovie (EachMovie website) data sets. The Jester data set contains the ratings on 100 jokes, among which 10 jokes are used to form the gauge set. The Jester data set has approximately 2,500,000 ratings on 100 items from 57,000 users. The rating scale ranges from −10 to 10, and continuous rating values were collected using the horizontal 'rating bar'. At the time of experimentation, not all of the Jester data were available to the authors, and therefore, only the available data on 17,720 users were used.

The MovieLens data were collected through the MovieLens web site for seven months by the GroupLens Research Project (GroupLens website). For the present investigation, the first data set was used. It consists of

Table 1
Experimental data

Data sets	Number of users	Number of items	Number of gauge items
Jester	17,720	100	10
MovieLens	244	118	5
EachMovie	4,849	245	10

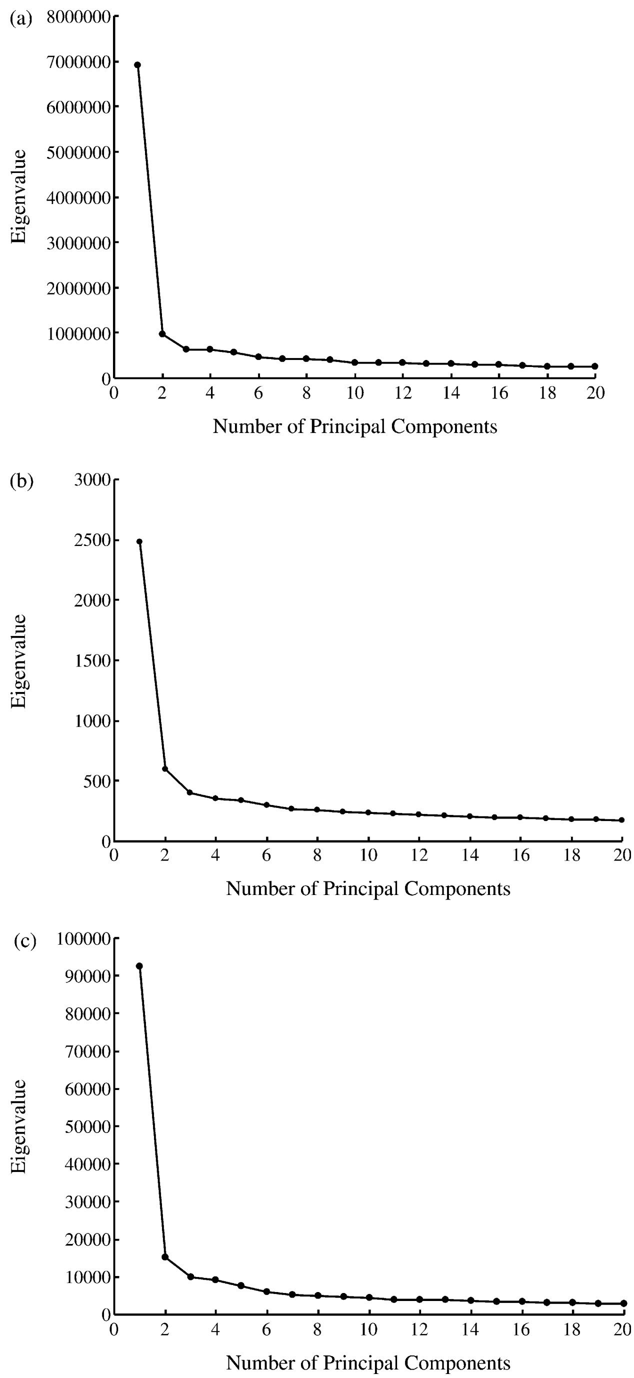

Fig. 2. Scree plots for the three data sets. (a) Jester data set, (b) MovieLens data set, (c) EachMovie data set.

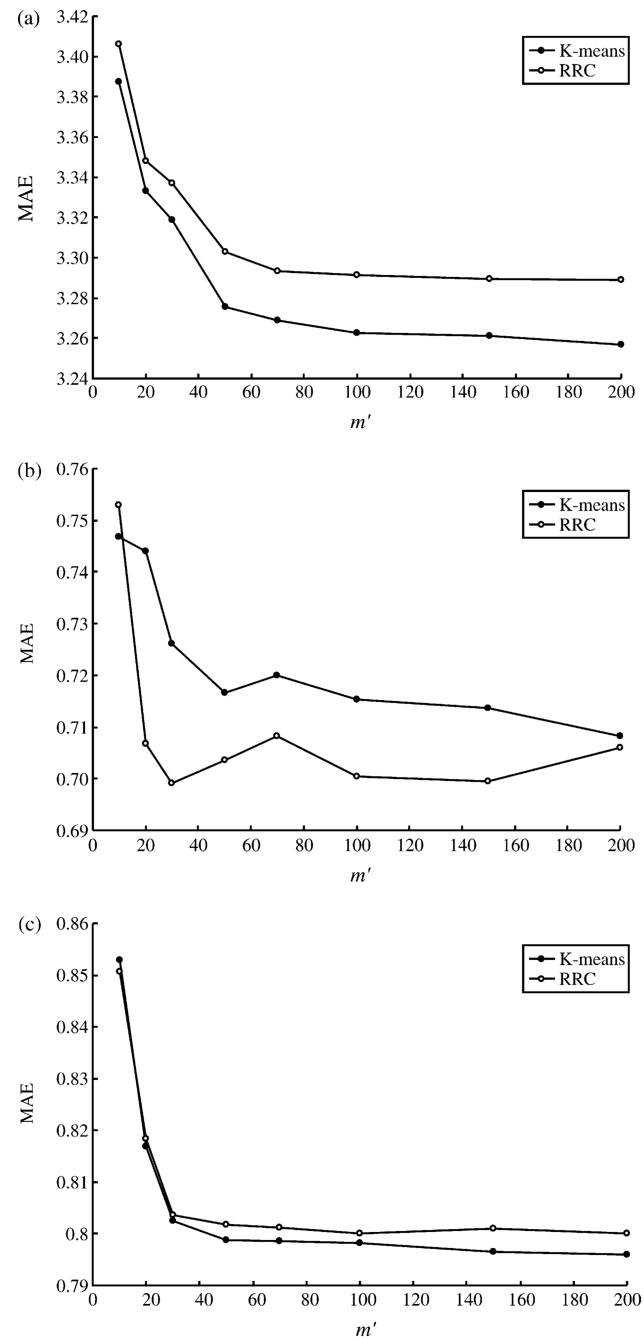

Fig. 3. MAE vs m' for three data sets. (a) Jester data set, (b) MovieLens data set, (c) EachMovie data set.

Table 2
Number of clusters

Data sets	Proposed approach		Eigentaste	
	RRC	*K*-means	RRC	*K*-means
Jester	52	61	52	57
MovieLens	52	54	52	49
EachMovie	52	57	52	58

100,000 ratings of 943 users on 1682 movies using the 1–5 integer rating scale.

EachMovie data set (EachMovie website) was collected by the Compaq Systems Research Center for 18 months to experiment with a collaborative filtering algorithm. EachMovie data set consists of 2,811,983 ratings on 1628 different movies (films and videos) by 72,916 users. Scores (i.e. ratings) range from 0 to 1 in increments of 0.2. In the present study, this is re-scaled to have equivalent 0–5 integer scores for simplicity.

Table 3
Prediction accuracy

Data sets	Approach	Clustering method	
		RRC	*K*-means
Jester	Proposed	3.29 (0.165)	3.28 (0.164)
	Eigentaste	3.65 (0.182)	3.78 (0.189)
MovieLens	Proposed	0.73 (0.184)	0.72 (0.179)
	Eigentaste	0.85 (0.211)	0.86 (0.215)
EachMovie	Proposed	0.80 (0.160)	0.80 (0.160)
	Eigentaste	0.84 (0.168)	0.85 (0.170)

(●): Normalized MAE.

Table 4
Analysis of variance for normalized MAE

Source	DF	SS $\times 10^4$	MS $\times 10^4$	*F*	*P*
Data	2	22.372	11.186	312.16	0.003
Approach	1	12.607	12.607	351.84	0.003
Clustering	1	0.041	0.041	1.14	0.398
Data × approach	2	2.535	1.267	35.37	0.027
Data × clustering	2	0.062	0.031	0.86	0.538
Approach × clustering	1	0.301	0.301	8.40	0.101
Error	2	0.072	0.036		
Total	11	37.989			

In the MovieLens and EachMovie data sets, the so-called gauge set is not defined. Since a gauge set is required for the *Eigentaste* algorithm, it was constructed for each of the two data sets according to the following procedures.

1. Arrange the item columns of the user-item matrix in descending order of the number of actual ratings.
2. Remove those rows that have a blank cell in the first item column from the user-item matrix. This generates a gauge item column.
3. Repeat Step 2 sequentially for the remaining item columns. In the course of Step 3, a certain item column may lose entries and become a column with unrated items only. Step 3 continues until the number of gauge item columns is close to 10 (i.e. the number of gauge items in the Jester data set), while the number of remaining users is maintained to be at least 200.
4. The resulting user-item matrix with the gauge items is used for the experiment.

Table 1 shows the characteristics of the three user-item matrices used for the present investigation. Note that, for the Movielens or EachMovie data set, the number of items is less than that of the original data set since those columns with unrated entries only are eliminated in the course of generating gauge items.

For the experiment, each user-item matrix was randomly divided into two sub-matrices. One is the estimation matrix which consists of 80% of the rows (i.e. users) of the user-item matrix, and the other is the test matrix of the remaining rows. Each row in the test matrix is regarded as a new user, and to compare the prediction accuracy of the *Eigentaste* and proposed approach, about 20% of the actual ratings of each new user were assumed to be missing, and the actual and the corresponding predicted ratings were compared to determine the prediction accuracy of each approach.

In the initial phase of the experiment, the iterative PCA was performed on the estimation matrix and the optimal number of PC's was determined. It is found that 2 is the optimal number of PC's for all three data sets as shown by the scree plots in Fig. 2 (also refer to Section 3.1 for the meaning of eigenvalue σ_{ii}^2).

Next, the appropriate number of clusters for the K-means clustering as well as the RRC algorithm was determined as shown in Table 2 based on the method described in Section 3.2.

4.2. Measure of prediction accuracy

MAE is used to evaluate the prediction ability of each method. The MAE is frequently used in the CF literature, and represents the difference between the predicted and

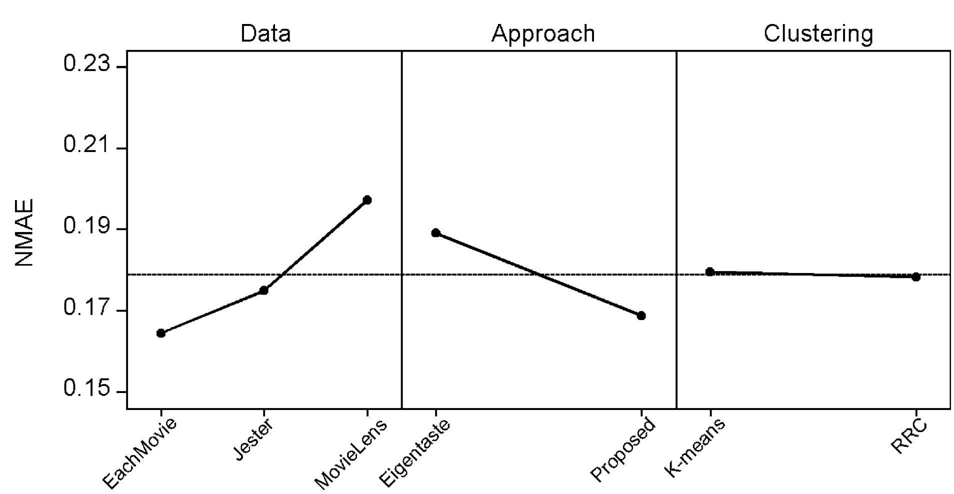

Fig. 4. Main effect plots.

the actual ratings of users as follows (Goldberg et al., 2001; Resnick, et al., 1994).

$$\mathrm{MAE} = \frac{1}{T}\sum_{t=1}^{T}|x_t - \hat{x}_t| \quad (3)$$

where, T is the number of actual ratings of the new users assumed to be missing in the test matrix, and x and $\hat{x}$ are the actual and the predicted ratings, respectively. In addition, the following normalized MAE (NMAE) is often used (Goldberg et al., 2001).

$$\mathrm{NMAE} = \frac{\mathrm{MAE}}{\text{Range of rating scale}}$$

4.3. Results

The proposed and *Eigentaste* algorithms, combined with the two clustering methods, are compared in terms of MAE (or NMAE). For the proposed method, $(m'-1)$ rows of the existing user-item matrix are used to predict the missing ratings for the new user. In the experiment, m' is varied such that $m'=10$, 20, 30, 50, 70, 100, 150 and 200. For the three data sets, Fig. 3(a)–(c) show how MAE values change with respect to m' for the Jester, MovieLens, and EachMovie data sets, respectively. Note that MAE values start to stabilize when m' is approximately equal to 50 for all three data sets regardless of the clustering method.

The MAE and NMAE values are summarized in Table 3 with respect to the data set, approach taken (Proposed vs *Eigentaste*), and clustering method. For the proposed method, MAE (or NMAE) values when $m'=50$ are included. From Table 3, we observe that the two clustering methods perform similarly regardless of the data set and approach taken, and more importantly, the proposed approach yields consistently better results (i.e. smaller MAE values) than *Eigentaste*.

To assess the effects of various parameters on the prediction accuracy in a more succinct manner, the analysis of variance (ANOVA) technique is applied to the NMAE. The experimental setting may be regarded as a full factorial design (Montgomery, 1999), the factors of which consists of data (with 3 levels of Jester, MovieLens, and EachMovie), approach (with 2 levels of Proposed and *Eigentaste*), and clustering (with 2 levels of RRC and K-means algorithm).

Table 4 shows the ANOVA table in which the three-way interaction effect (i.e. data × approach × clustering) is assumed to be negligible. Note that the p-values for main effects data and approach as well as for interaction effect data × approach are 'small', and therefore, those effects are considered statistically significant (Montgomery, 1999). This is also shown in Figs. 4 and 5, which, respectively, show the three main effects and three interaction effects. Fig. 4 shows that the NMAE for the MovieLens data is higher than the others. This is believed to be due to the relatively small size of the corresponding user-item matrix

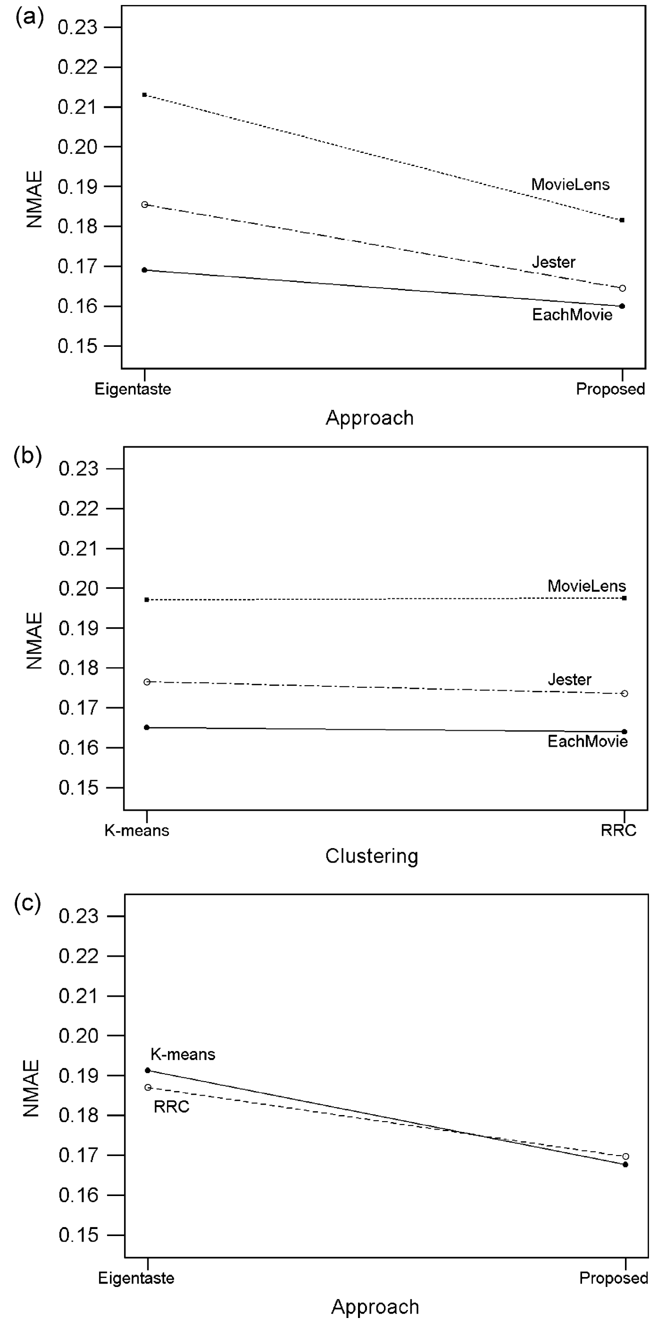

Fig. 5. Interaction plots. (a) Interaction effect: data × approach. (b) Interaction effect: data × clustering. (c) Interaction effect: approach × clustering.

remained after forming a gauge set (See Table 1). The data × approach interaction plot (see Fig. 5(a)) indicates that the proposed approach performs better than *Eigentaste*, although the NMAE difference between the two approaches differs from data set to data set.

Since the degree of freedom for the error term in Table 4 is small, main effect clustering and interaction effects data × clustering were pooled into the error term and a new

ANOVA was performed, resulting in the same conclusions as before.

Although the above statistical analysis indicates that the proposed approach performs better than *Eigentaste* regardless of the data set and clustering method considered, the MAE differences between the two approaches may not be considered practically significant. What is important, however, is that the prediction accuracy of the proposed approach does not deteriorate even without a gauge set.

5. Conclusions

An iterative PCA approach to CF is developed, and is compared to *Eigentaste* in terms of MAE (or NMAE) using the Jester, MovieLens, and EachMovie data sets. Experimental results show that the proposed approach performs consistently better than *Eigentaste* regardless of the data set and clustering method employed, although the differences in MAE between the two approaches may not be practically significant. This is an encouraging result since the proposed approach, unlike *Eigentaste*, does not require a gauge set to be defined. In other words, the performance of the proposed approach does not deteriorate even without a gauge set, and therefore, may be considered as a useful alternative when defining a gauge set is neither possible nor practical.

For recommending items to a new user, the proposed approach utilizes data from previous users (about 50 users) for an iterative PCA. The time required for this step varies from less than a second to a few seconds depending on the proportion of unrated items, stopping criterion (i.e. c in Eq. (2)), etc. It may be a fruitful area of future research to develop an efficient method for updating SVD results whenever a new user enters the system.

Acknowledgements

We wish to thank Professor Ken Goldberg at University of California, Berkeley for providing us with the Jester data set.

References

Adams, E., Walczak, B., Vervaet, C., Risha, P. G., & Massart, D. L. (2002). Principal component analysis of dissolution data with missing elements. *International Journal of Pharmaceutics*, *234*(1–2), 169–178.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification*. New York: Wiley.

EachMovie website: http://research.compaq.com/SRC/eachmovie/

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

Golub, G. H., & Van Loan, C. F. (1996). *Matrix computations* (3rd ed.). MD: Johns Hopkins University Press.

GroupLens website: http://www.grouplens.org/

Johnson, R. A., & Wichern, D. W. (2002). *Applied multivariate statistical analysis* (5th ed.). NJ: Prentice-Hall.

MacQueen, J. B. (1967). Some methods for classification and analysis of multivariate observations. Proceedings of the 5th Berkeley symposium on mathematical statistics and probability (pp. 281–297).

Matlab Statistics Toolbox: http://www.mathworks.com/access/helpdesk/help/toolbox/stats/multiv18.html

Montgomery, D. C. (1999). *Design and analysis of experiments* (5th ed.). New York: Wiley.

MovieLens website: http://www.movielens.umn.edu

Resnick, P., Iacocou, N., Suchak, M., Berstrom, P., Riedl, J. (1994). Grouplens: an open architecture for collaborative filtering of netnews. Proceedings of the ACM conference on computer support cooperative work (pp. 175–186).

Sarwar, B. M., Karypis, G., Konstan, J. A., Riedl, J. T. (2000). Application of dimensionality reduction in recommender System-A case study. Proceedings of ACM WebKDD workshop.

Walczak, B., & Massart, D. L. (2001). Dealing with missing data part I. *Chemometrics and Intelligent Laboratory Systems*, *58*(1), 15–27.