International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 34, No. 1, February, 2020, pp. 59-65
Journal homepage: http://iieta.org/journals/ria

Improving Multi-class Co-Clustering-Based Collaborative Recommendation Using Item Tags

Anita Gholami, Yahya Forghani*

Mashhad Branch, Islamic Azad University, Mashhad 9187147578, Iran

Corresponding Author Email: yforghani@mshdiau.ac.ir

https://doi.org/10.18280/ria.340108

Received: 4 October 2019
Accepted: 20 December 2019

Keywords:
recommender system, collaborative filtering, multi-class Co-Clustering (MCoC), rating vector, tag vector

ABSTRACT

Multi-class Co-Clustering (MCoC)-based recommendation system is a method for recommending favorite items to users. It, firstly, groups items and users in a way that users have common interests and their favored items are put in the same group. Then, to estimate unrated items of each group, for each group, an independent collaborative filtering approach is implemented. MCoC-based recommendation system uses rating vectors to determine similar items and to group them together. We expect that the ratings of two items with the same tag be closer than two items with different tags. Therefore, tags can also be used to determine similar items for co-clustering. The purpose of this study is to enhance the performance of co-clustering to increase the accuracy of MCoC-based recommendation system by using item tags in addition to item rating vectors. Our experiments show that mean absolute error of our proposed method is much less than that of MCoC-based recommendation system.

1. INTRODUCTION

Recommendation systems are today unavoidable for web users. The aim of recommendation systems is suggesting relevant items to users. Three main paradigms of recommendation systems are content-based recommender systems [1], collaborative filtering-based recommender systems [2, 3], and hybrid approaches [4]. Collaborative filtering methods are divided into memory-based and model-based methods. Memory-based methods directly work with user-item ratings, do not make a model, and work based on nearest neighbors' search algorithm to determine similar users or items [5]. Model-based algorithms learn a model on the basis of user-item ratings and then the model is used to estimate unrated items [6]. Two traditional methods to determine nearest neighbors in memory-based methods are Pearson correlation and cosine similarity. These methods have cold start problem. To alleviate this problem, a new similarity criterion was proposed [7, 8]. One can also cluster items or users to specify nearest items or nearest items [9]. M-distance is another criteria to determine similar items [10]. The M-distance of two items is the difference between the mean ratings of each of these two items. The mean ratings on each item is computed in the training phase once. M-distance compares two scalars, i.e. two means, to determine their similarity, while in cosine similarity and Pearson correlation, two vectors of ratings are compared. Therefore, computing M-distance is less time-consuming than Pearson correlation and cosine similarity. In the VM-distance method, in addition to the mean of ratings, the variance of ratings also is used for better specification of similar items [5]. Slope-one [11] is a simple linear regression model for recommendation system. The incremental Slope-one [12] do regression incrementally for online situation.

Some collaborative filtering methods first group users. Then, the same suggestions are given to each group of users. But it is almost impossible to find group of users with common interests on the whole of items. Strictly speaking even if a group of users have similar interests on one set of items, their interests may be not the same on another set of items. Therefore, we cannot estimate every unrated items of this group of users on the basis of only this group of users. Therefore, we must group users on a subset of items. In other words, both users and items must be incorporated in grouping. Such method is named co-clustering [13]. Co-clustering groups users based on their ratings on items in the same group, and groups items based on the ratings of the users in the same group. Multi-class Co-Clustering (MCoC) is a co-clustering method which allows each user or item becomes the members of several groups [14]. To estimate unrated items of each group in MCoC, for each group, an independent collaborative filtering approach such as slope-one approach or matrix factorization approach [15] is implemented [16]. Co-clustering can also alleviate cold start problem of recommendation systems [13].

In some datasets, items have tags. For example, in Movielens dataset, the tag of an item is the gender of a film. We expect that the ratings of two items with the same item tag be closer than two items with different item tags in such datasets. Therefore, item tags can also be used for item similarity weight determination and item grouping in co-clustering. One of the disadvantages of MCoC-based collaborative recommendation model is that it doesn't use item tags for co-clustering. Therefore, in this paper, MCoC-based collaborative recommendation model is reformulated to use item tags for a more accurate co-clustering. After co-clustering, the ratings of unrated items of each group are estimated based-on a collaborative filtering method such as on the basis of users and items of that group. Since, each user is a member of several groups, different rating estimations for an unrated item are obtained. The MCoC-based collaborative recommendation system reports the weighted average of estimated ratings as final estimated rating of that unrated item. In this paper, an improved weighting formula is also proposed to enhance the final estimation. Our experiments on real datasets show that mean absolute error of the proposed method is less than the traditional MCoC-based collaborative recommendation system.

In continue, in section 2, The MCoC-based collaborative recommendation method is explained in detail. In section 3, our proposed method is presented. In section 4, the results of experiments are presented and in section 5 the conclusion is drawn.

2. MCOC-BASED RECOMMENDER SYSTEM

In the MCoC-based collaborative recommendation system, firstly, items and users are grouped in a way that the users have common tastes and their favored items are put in the same group. Then, ratings of unrated items of each group are estimated based-on a collaborative filtering method on the basis of users and items of that group.

Consider user-item matrix $T \in R^{n \times m}$, which includes the ratings of n users to at most m items. T_{ij} is the rating given by i-th user to j-th item. If T_{ij} is equal to zero, it means that the user rating for this item is unclear. The rating matrix T is a sparse matrix, namely the most of ratings are zero. In the MCoC-based collaborative recommendation system, users and items are grouped into c groups. Each user or item becomes the member of k groups. Since users and items can be grouped together, this method is called co-clustering. The results of this grouping or clustering are recorded in the membership degree matrix $P \in \{0,1\}^{(n+m) \times c}$. If i-th object, which is a user or an item, belongs to j-th group, then $P_{ij} = 1$, otherwise $P_{ij} = 0$. Meanwhile, the sum of each row of P or the sum of membership degrees of the i-th object in groups denoted by P_i is equal to k, because each user or item is member of k groups. Indeed, the matrix P, is the join of two sub-matrices $Q \in \{0,1\}^{n \times c}$ (the membership degree matrix of users in groups) and $R \in \{0,1\}^{m \times c}$ (the membership degree matrix of items in groups), namely:

$$P = \binom{Q}{R} \qquad (1)$$

For co-clustering, the similarity or relationship between users and items must be computed. If the similarity or relationship between two users, two items, or a user and an item is large enough, then those similar objects must be grouped together.

2.1 Grouping a user and an item together

The larger T_{ij} is, the greater is the relationship between i-th user and j-th item. This is the relationship of interest. Therefore, the relationship between i-th user and j-th item is determined by the rating T_{ij}. We want the user and item that have more relationship with each other to be in the same group. For this purpose, the following model is used:

$$min_{Q,R} \sum_{i=1}^{n} \sum_{j=1}^{m} \left(\left\| \frac{q_i}{\sqrt{D_{ii}^{row}}} - \frac{r_j}{\sqrt{D_{jj}^{col}}} \right\|^2 T_{ij} \right) \qquad (2)$$

where, q_i is i-th row of the matrix Q or the membership degree of i-th user in groups, and r_j is j-th row of the matrix R or the membership degree of j-th item in groups. D_{ii}^{row} is the sum of the ratings of i-th user, i.e.

$$D_{ii}^{row} = \sum_{j=1}^{m} T_{ij} \qquad (3)$$

and D_{jj}^{col} is the sum of the ratings of different users for j-th item, i.e.

$$D_{jj}^{col} = \sum_{i=1}^{n} T_{ij} \qquad (4)$$

In fact, D_{jj}^{col} and D_{ii}^{row} are used for normalization in model (2). The model (2) can be written equivalently as follows:

$$\begin{gathered} min_{Q,R} \sum_{i=1}^{n} \|q_i\|^2 + \\ \sum_{j=1}^{m} \|r_i\|^2 - \sum_{i=1}^{n} \sum_{j=1}^{m} \frac{2q_i^T r_j T_{ij}}{\sqrt{D_{ii}^{row} D_{jj}^{col}}} \\ = Tr(Q^T Q + R^T R - Q^T S R) \\ = \mathrm{Tr}\left((\mathrm{Q^T}\ \mathrm{R^T}) \begin{pmatrix} \mathrm{I_n} & -\mathrm{S} \\ -\mathrm{S^T} & \mathrm{I_m} \end{pmatrix} \begin{pmatrix} \mathrm{Q} \\ \mathrm{R} \end{pmatrix} \right) = Tr(P^T M_{UI} P) \end{gathered} \qquad (5)$$

where,

$$S = (D^{row})^{-\frac{1}{2}} T \, (D^{col})^{-\frac{1}{2}} \qquad (6)$$

$$M_{UI} = \begin{pmatrix} I_n & -S \\ -S^T & I_m \end{pmatrix} \qquad (7)$$

and I_n is $n \times n$ identity matrix, D^{row} is a diagonal matrix whose main diagonal elements are D_{ii}^{row} $(i = 1,2,..,n)$, and D^{col} is a diagonal matrix whose main diagonal elements are D_{ii}^{col} $(i = 1,2,..,m)$.

2.2 Grouping two users together

The following model is used to group similar users together:

$$min_Q \sum_{i=1,j=1}^{n} \left(\left\| \frac{q_i}{\sqrt{D_{ii}}} - \frac{q_j}{\sqrt{D_{jj}}} \right\|^2 W_{ij} \right) \qquad (8)$$

where, W_{ij} is similarity weight between i-th user and j-th user which is computed as follows:

$$W_{ij} = exp\left(\frac{-\|T_i - T_j\|^2}{\sigma^2} \right) \qquad (9)$$

Meanwhile, D_{ii} which is used for normalization in the model (8) is as follows:

$$D_{ii} = \sum_{j=1}^{m} W_{ij} \qquad (10)$$

The model (8) can be written equivalently as follows:

$$min_Q \, Tr(Q^T L_Q Q) = Tr\left((Q^T\ R^T) \begin{pmatrix} L_Q & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} Q \\ R \end{pmatrix} \right) \qquad (11)$$

where,

$$L_Q = D - W \qquad (12)$$

and D is a diagonal matrix of which main diagonal elements are D_{ii} $(i = 1,2,.., n)$. The model (11) is equivalent to

$$min_P Tr(P^T M_{UU} P) \tag{13}$$

where,

$$M_{UU} = \begin{pmatrix} L_Q & 0 \\ 0 & 0 \end{pmatrix} \tag{14}$$

2.3 Grouping two items together

Similarly, the following model is used to group similar items together:

$$min_R \sum_{i=1,j=1}^{m} \left(\left\| \frac{r_i}{\sqrt{\widetilde{D}_{ii}}} - \frac{r_j}{\sqrt{\widetilde{D}_{jj}}} \right\|^2 \widetilde{W}_{ij} \right) = Tr(P^T M_{II} P) \tag{15}$$

$$M_{II} = \begin{pmatrix} 0 & 0 \\ 0 & L_R \end{pmatrix} \tag{16}$$

$$L_R = \widetilde{D} - \widetilde{W} \tag{17}$$

where, $\widetilde{W}_{ij}$ is the similarity weight between i-th item and j-th item which is computed as follows:

$$\widetilde{W}_{ij} = exp\left(\frac{-\|T_{:,i} - T_{:,j}\|^2}{\sigma^2} \right) \tag{18}$$

where, $T_{:,i}$ is i-th column of the rating matrix T or the ratings of i-th item. Moreover, $\widetilde{D}_{jj}$ which is used for normalization in the model (15) is as follows:

$$\widetilde{D}_{jj} = \sum_{i=1}^{n} \widetilde{W}_{ij} \tag{19}$$

The matrix $\widetilde{D}$ is a diagonal matrix whose main diagonal elements are $\widetilde{D}_{jj}$ $(j = 1,2,.., m)$. By combining the objective function of three models (5), (13) and (15), the final model of the MCoC-based collaborative recommendation system is obtained which is as follows:

$$\begin{aligned} min_{Q,R} & \sum_{i=1}^{n} \sum_{j=1}^{m} \left(\left\| \frac{q_i}{\sqrt{D_{ii}^{row}}} - \frac{r_j}{\sqrt{D_{jj}^{col}}} \right\|^2 T_{ij} \right) \\ & + \sum_{i=1,j=1}^{n} \left(\left\| \frac{q_i}{\sqrt{D_{ii}}} - \frac{q_j}{\sqrt{D_{jj}}} \right\|^2 W_{ij} \right) \\ & + \sum_{i=1,j=1}^{m} \left(\left\| \frac{r_i}{\sqrt{\widetilde{D}_{ii}}} - \frac{r_j}{\sqrt{\widetilde{D}_{jj}}} \right\|^2 \widetilde{W}_{ij} \right) \end{aligned} \tag{20}$$

The model (20) is equivalent to the following model:

$$min_P Tr(P^T M P) \tag{21}$$

where,

$$M = M_{UI} + M_{UU} + M_{II} = \begin{pmatrix} I_n + L_Q & -S \\ -S^T & I_m + L_R \end{pmatrix} \tag{22}$$

The MCoC-based collaborative recommendation model has also constraint: each item or each user must be the member of k groups of c groups. The MCoC-based collaborative recommendation model is as follows:

$$\begin{aligned} & min_P Tr(P^T M P) \\ & subject\ to \begin{cases} P \in R^{(m+n)\times c}, \\ P \geq 0, \\ P1_c = 1, \\ |P_i| = k. \end{cases} \end{aligned} \tag{23}$$

where, $|P_i|$ is the cardinality of P_i. The constraints $P \geq 0$ and $P1_c = 1$ force each entry of P stay in the range [0,1], and the sum of each row of P is equal to 1. Solving the model (23) is difficult. Therefore, an approximate method is used to solve this model [14]. In this approximation method, Q and R which construct P, are first taken into a spectral space such that relationship or similarity weight between each two users, relationship or similarity weight between each two items, and relationship between each user and each item in that space, is maintained. To this end, the objective function of the model (23) must be minimized. Then, clustering in spectral space is performed in such a way that the constraints of the model (23) are satisfied. To map P into a spectral space where the relationship between users and items is preserved, the following model is used:

$$\begin{aligned} & min_X Tr(X^T M X) \\ & subject\ to\ X^T X = I. \end{aligned} \tag{24}$$

where, $X \in R^{(m+n)\times r}$ is r-dimensional spectral axis. This new space has several features. Its dimension is r which is less than or equal to the dimension of P_i's which are equal to c. r is greater than k, and is determined by the user. As stated earlier, the relationship or similarity weight between users and items in the new spectral space is preserved. Therefore, to solve problem (23), after solving the model (24), we only have to cluster the matrix X in such a way that the constraints of problem (23) also is satisfied. For this purpose, the fuzzy c-means clustering method is used to cluster the rows of the matrix X into c clusters. The fuzzy c-means model is as follows:

$$\begin{aligned} & min_{P,V} \sum_{i=1}^{m+n} \sum_{j=1}^{c} P_{ij}^{l} \|X_i - v_j\|^2 \\ & subject\ to \begin{cases} \forall i: \sum_{j=1}^{c} P_{ij} = 1, \\ \forall i,j: P_{ij} \geq 0. \end{cases} \end{aligned} \tag{25}$$

where, l is fuzziness parameter. To solve the fuzzy c-means problem, the values of P and V are updated as follows until convergence:

$$P_{ij} = \frac{1}{\sum_{t=1}^{c} \left(\frac{\|X_i - v_j\|}{\|X_i - v_t\|} \right)^{2/(l-1)}} \tag{26}$$

$$v_j = \left(\sum_{i=1}^{m+n} P_{ij}^{l} X_i \right) / \sum_{i=1}^{m+n} P_{ij}^{l}, \tag{27}$$

where, $i = 1, 2, \ldots, (n + m)$ and $j = 1, 2, \ldots, c$. The matrix P contains the membership degrees of users and items in different groups. The constraints of the problem (23) have not been satisfied yet. To satisfy these constraints, in each row of matrix P, k larger values are maintained, and the remaining values are put to zero. Eventually, each row of matrix P are normalized. In this way, the constraints of problem (23) are also satisfied and the problem (23) is solved. Algorithm 1 summarizes the approach used for solving the model (23).

Algorithm 1. An approach for solving the model (23).

```
Input:
•  The matrix $M$.
•  The number of groups: $c$;
•  Each user or item is the member of how many groups: $k$;
•  The dimension of spectral space: $r$;
Output:
    The matrix $P$ which is join of two sub-matrices $Q \in \{0,1\}^{n\times c}$ (the membership degree matrix of users in groups) and $R \in \{0,1\}^{m\times c}$ (the membership degree matrix of items in groups).
Begin
•  Solve the model (24) and obtain optimal value of $X$: The solution is $r$ Eigen vectors of $r$ least Eigen values of the matrix $M$ [14].
•  Cluster $X$ to c clusters using the model (25) and obtain optimal membership values $P$ and cluster centers $v$: Compute iteratively Eq. (26) and Eq. (27) until convergence.
End
```

Finally, grouping results are used to predict unknown values of the rating matrix T. Assume that $Pr(u_i, y_j, t)$ is the rating of i-th user (u_i) for j-th item (y_j), which is estimated based on some collaborative filtering methods such as the slope-one method, based solely on users and items of t-th group. Since, each user is a member of several groups, different rating estimations for an unrated item are obtained. The MCoC-based collaborative recommendation system proposes the weighted average of estimated ratings as final estimated rating of that unrated item. In other words, final estimated rating of i-th user for j-th item are calculated as follows:

$$\tilde{T}_{ij} = \begin{cases} \sum_t Pr\,(u_i\,, y_j, t) P_{it} & if\ u_i\ and\ y_j\ are\ members\ of \\ & at - least\ one\ common\ group \\ 0 & otherwise. \end{cases} \tag{28}$$

where, P_{it} is the membership degree of i-th user in t-th group.

3. OUR PROPOSED MODEL (EXTENDED MCOC-BASED RECOMMENDATION MODEL)

The traditional MCoC-based collaborative recommendation model, i.e. the model (23), uses rating vectors of items to determine similar items and to group them together. We expect that the ratings of two items with the same item tags be closer than two items with different item tags. Therefore, item tags can also be used for item similarity weight determination and item grouping in co-clustering. In this paper, item tags are also used for a more accurate items grouping in the MCoC-based collaborative recommendation model. To this end, we propose a model with the following objective function:

$$\min_{q,r} \quad \sum_{i=1}^{n} \sum_{j=1}^{m} \left(\left\| \frac{q_i}{\sqrt{D_{ii}^{row}}} - \frac{r_j}{\sqrt{D_{jj}^{col}}} \right\|^2 T_{ij} \right) + \sum_{i=1,j=1}^{n} \left(\left\| \frac{q_i}{\sqrt{D_{ii}}} - \frac{q_j}{\sqrt{D_{jj}}} \right\|^2 W_{ij} \right) \tag{29}$$

$$+ \sum_{i=1,j=1}^{m} \left(\left\| \frac{r_i}{\sqrt{\bar{D}_{ii}}} - \frac{r_j}{\sqrt{\bar{D}_{jj}}} \right\|^2 \bar{W}_{ij} \right)$$

where, $\bar{W}_{ij}$ is the similarity weight between i-th item and j-th item which is as follows:

$$\bar{W}_{ij} = exp\left(\frac{-\left\| \begin{pmatrix} T_{:,i} \\ E_i \end{pmatrix} - \begin{pmatrix} T_{:,j} \\ E_j \end{pmatrix} \right\|^2}{\sigma^2} \right) \tag{30}$$

where, E_i is the tag vector of i-th item. Each element of E_i is equal to zero or one. If k-th element of E_i is equal to one, it means that i-th item has k-th tag (k-th gender in the Movielens dataset). In fact, the main difference between our proposed model and the traditional MCoC-based collaborative recommendation model, i.e. the model (23), is the use of $\bar{W}_{ij}$ in our proposed model instead of $\tilde{W}_{ij}$ in the model (23). As can be seen, $\bar{W}_{ij}$ uses tag vector of the items, in addition to the ratings vector of items, to calculate the similarity weight of two items and grouping similar items together, which can lead to a enhance the co-clustering accuracy. Meanwhile, $\bar{D}_{jj}$ which is used for normalization in the model (29) is as follows:

$$\bar{D}_{jj} = \sum_{i=1}^{n} \bar{W}_{ij} \tag{31}$$

The objective function (29) is equivalent to $Tr(P^T \bar{M} P)$, where

$$\bar{M} = M_{UI} + M_{UU} + \bar{M}_{II} = \begin{pmatrix} I_n + L_Q & -S \\ -S^T & I_m + \bar{L}_R \end{pmatrix} \tag{32}$$

$$\bar{M}_{II} = \begin{pmatrix} 0 & 0 \\ 0 & \bar{L}_R \end{pmatrix} \tag{33}$$

$$\bar{L}_R = \bar{D} - \bar{W} \tag{34}$$

where, $\bar{D}$ is a diagonal matrix of which main diagonal elements are $\bar{D}_{jj}$ ($j = 1,2, \dots, m$). Therefore, our proposed model is as follows:

$$\begin{aligned} & min_P\ Tr(P^T \bar{M} P) \\ & subject\ to \begin{cases} P \in R^{(m+n)\times c}, \\ P \geq 0, \\ P1_c = 1, \\ |P_i| = k. \end{cases} \end{aligned} \tag{35}$$

To solve the model (35), we use the same method used in [14] to solve the model (23) which was expressed in previous section. Finally, grouping results are used to predict unknown values of the rating matrix T. Assume that $Pr(u_i, y_j, t)$ is the rating of i-th user (u_i) for j-th item (y_j), which is estimated based on some collaborative filtering methods such as the slope-one method, based solely on users and items of t-th group. Since, each user is a member of several groups, different rating estimations for an unrated item are obtained. The MCoC-based collaborative recommendation system proposes the weighted average of estimated ratings as final estimated rating of that unrated item, i.e. Eq. (28). The weight or confidence level of the estimated rating of i-th user for j-th item based solely on users and items of t-th group in Eq. (28) is the membership degree of i-th user in t-th group, i.e. P_{it}, while membership degree of *j*-th item in *t*-th group, i.e. P_{jt}, affects also the confidence level of the rating estimation. Therefore, the following equation is proposed for determining the final rating of *i*-th user for *j*-th item:

$$\tilde{T}_{ij} = \begin{cases} \frac{\sum_t Pr\,(u_i, y_j, t) P_{it} \times P_{jt}}{\sum_t P_{it} \times P_{jt}} & if\ u_i\ and\ y_j\ are\ members\ of \\ & at-least\ one\ common\ group \\ 0 & otherwise. \end{cases} \quad (36)$$

where, the denominator is used to normalize the weights in Eq. (36). Finally, we can summarize our proposed algorithm to solve the proposed model (29) as algorithm 2.

Algorithm 2. The proposed algorithm for solving our proposed model (29).

```
Input:
•  User-item matrix which contain both rated and unrated items: $T$;
•  Item tags: $E$;
•  The number of groups: $c$;
•  Each user or item is the member of how many groups: $k$;
•  The dimension of spectral space: $r$;
Output:
    New User-item matrix with estimated rates of unrated items: $\tilde{T}$;
Begin
•  Compute $\overline{W}$ using Eq. (30).
•  Compute $\overline{D}$ using Eq. (31).
•  Compute $M_{UI}$ using Eq. (7).
•  Compute $M_{UU}$ using Eq. (14).
•  Compute $\overline{L}_R$ using Eq. (34).
•  Compute $\overline{M}_{II}$ using Eq. (33).
•  Compute $\overline{M}$ using Eq. (32).
•  Solve the model (35) using algorithm 1 for the inputs $\overline{M}$, $c$, $k$, $r$ to obtain optimal $P$.
•  Use Eq. (36) to estimate unrated items and to obtain $\tilde{T}$.
End
```

4. EXPERIMENTAL RESULTS

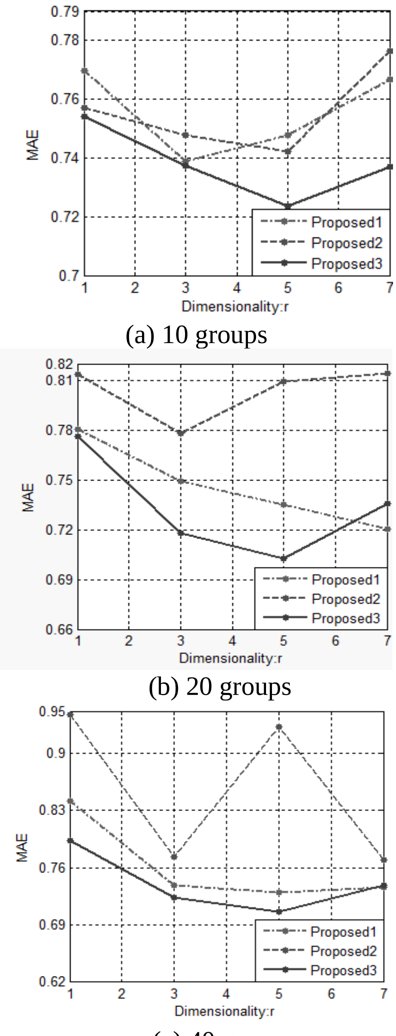
Our proposed system uses both rating vectors of items and tag vectors to determine similar items and to group them together. We name this system as "proposed3". To see the effect of using each of rating vectors and tag vectors solely for determining similar items in our proposed system, we propose two other systems. In the first one, we use only rating vectors for determining similar items, namely we use $\overline{W}_{ij} = exp\left(\frac{-\|T_{:,i}-T_{:,j}\|^2}{\sigma^2}\right)$ instead of Eq. (30). We name this system as "proposed1". In the second proposal, we use only tag vectors for determining similar items, namely we use $\overline{W}_{ij} = exp\left(\frac{-\|E_i-E_j\|^2}{\sigma^2}\right)$ instead of Eq. (30). We name this system as "proposed2".

The effectiveness of our proposed methods is evaluated by using some experiments on two real datasets, i.e. Movielens-100k and Movielens (ml-latest-small). MovieLens-100K and Movielens (ml-latest-small) are movie rating datasets. MovieLens-100K consists of 100,000 ratings (1-5) from 943 users on 1,682 movies. Movielens (ml-latest-small) consists of 100,004 ratings (1-5) from 671 users on 9,125 movies. Each user has rated at least 20 movies. The genres list of each film constructs its tag vector. Eighteen different genres are Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, Western. Therefore, tag vector is an 18-dimensional vector. When i-th element of the tag vector of a film is 1, it means that the film is in i-th genre.

Each dataset is randomly divided into an 80 percent training part and a 20 percent testing part. We run our experiment on the training set and evaluate on the testing set and report the mean absolute error (MAE). We use grid search to obtain the best values of parameters. We select the best value of the dimension of spectral space r from the set $\{1, 3, 5, 7\}$, and the best value of number of groups c from the set $\{10, 20, 40\}$. The number of groups to which a user or an item belongs, i.e. *k*, is set to *ceil(log(c))* as in [14]. Table 1 shows the best MAE of different recommendation systems. As it can be seen, the third proposed system has the best MAE. The third proposed model uses both tag vectors and rating vector to determine similar items. This means that none of rating vectors and tag vectors solely can co-cluster items and users well. Indeed, the model "proposed2" grouped films with similar genres together while film genres are not single feature of film. For example, animation is not a genre while we like to group animations with the same genres together. We have no film information else genres in tag vectors. Therefore, we must also use rating vector of films to determine similar films more accurately as in "proposed3". The model "proposed1" grouped films with similar rating vector together solely and ignore tag vectors, i.e. genres, to determine similar films. The MCoC-based collaborative recommendation system has the worst MAE. The reason is that the MCoC-based collaborative recommendation system which proposes the weighted average of estimated ratings as final estimated rating of that unrated item, i.e. Eq. (28), uses the membership degree of *i*-th user in *t*-th group, i.e. P_{it}, as the weight or confidence level of the estimated rating of i-th user for *j*-th item based solely on users and items of *t*-th group, while membership degree of *j*-th item in *t*-th group, i.e. P_{jt}, affects also the confidence level of the rating estimation. The MAE of "proposed3", which is the best of our proposed systems, is 2% less than that of the slope one, and much less than the MCoC-based collaborative recommendation system.

Table 1. The best MAE of different recommendation systems and their parameters

	Movielens-100k	**MovieLens (ml-latest-small)**
Slope one [11]	0.7280	0.6634
MCoC [14]	0.7581 *c*=10, *r*=3	0.9583 *c*=10, *r*=7
Proposed1	0.7204 *c*=20, *r*=7	0.6590 *c*=10, *r*=7
Proposed2	0.7421 *c*=10, *r*=5	0.6957 *c*=10, *r*=7
Proposed3	0.7029 *c*=20, *r*=5	0.6480 *c*=10, *r*=5

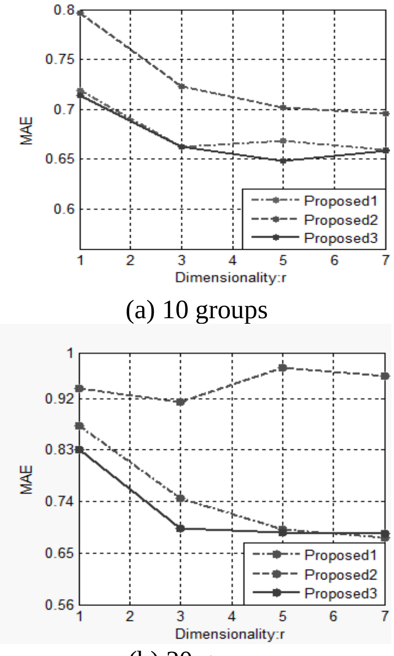


(a) 10 groups

(b) 20 groups

(c) 40 groups

Figure 1. The sensitivity of the proposed systems to the dimension of spectral space *r* for the dataset Movielens-100k



(a) 10 groups

(b) 20 groups

Figure 2. The sensitivity of the proposed systems to the dimension of spectral space *r* for the dataset Movielens (ml-latest-small)

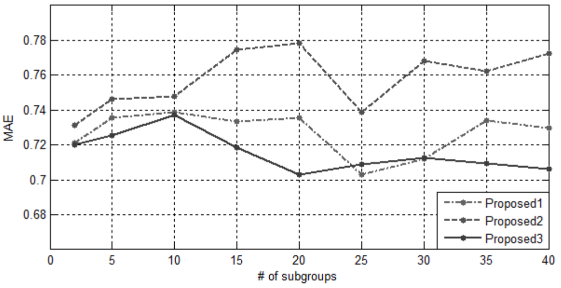


Figure 3. The sensitivity of the proposed systems to the number of groups *c* for the dataset Movielens-100k when *r*=3

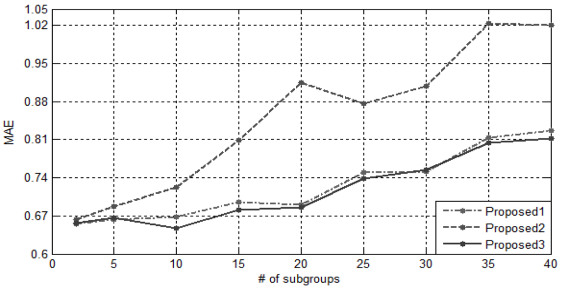


Figure 4. The sensitivity of the proposed systems to the number of groups *c* for the dataset Movielens (ml-latest-small) when *r*=3

Figure 1 and Figure 2 depict the sensitivity of the proposed systems to the dimension of spectral space *r*. As it can be seen, when *r* increases up to 5, MAE of the "proposed3" for each dataset decrease. Figure 3 and Figure 4 depict the sensitivity of the proposed systems to the number of groups *c* for *r*=3. As it can be seen, when *c* increases, MAE of the "proposed3" for the Movielens-100k dataset usually decrease while MAE of the "proposed3" for the Movielens (ml-latest-small) dataset usually increase. Therefore, the best value of the parameter *c* depends on dataset.

5. CONCLUSION

In the traditional MCoC-based collaborative recommendation system, firstly, items and users are grouped in a way that users have common interests and their favored items are put in the same group. The MCoC-based collaborative recommendation system uses rating vectors of items to determine similar items and to group them together. We expect that the ratings of two items with the same item tag be closer than two items with different item tags. Therefore, item tags can also be used to determine similar items for co-clustering. In this paper, item tags, in addition to rating vectors, were used for a more accurate items grouping in the MCoC-based collaborative recommendation model. Ratings of unrated items of each group are estimated based-on a collaborative filtering method on the basis of users and items of that group. Since, each user is a member of several groups, different rating estimations for an unrated item are obtained. The MCoC-based collaborative recommendation system proposes the weighted average of estimated ratings as final estimated rating of that unrated item. In this paper, an improved weighting formula was also proposed to enhance the final estimation. Our experiments on real datasets, i.e. Movielens-100k Movielens (ml-latest-small), and show that mean absolute error of our proposed method is about 2% less than that of the slope one, and much less than that of the traditional MCoC-based collaborative recommendation system. In future, we extend the MCoC-based collaborative recommendation system for a context-based recommendation.

REFERENCES

[1] Cruz, A.F.T., Coronel, A.D. (2020). Towards developing a content-based recommendation system for classical music. In Information Science and Applications, 621: 451-462. https://doi.org/10.1007/978-981-15-1465-4_45

[2] Nassar, N., Jafar, A., Rahhal, Y. (2020). A novel deep multi-criteria collaborative filtering model for recommendation system. Knowledge-Based Systems, 187: 104811. https://doi.org/10.1016/j.knosys.2019.06.019

[3] Ramakrishnan, G., Saicharan, V., Chandrasekaran, K., Rathnamma, M.V., Ramana, V.V. (2020). Collaborative filtering for book recommendation system. Soft Computing for Problem Solving, 1057: 325-338. https://doi.org/10.1007/978-981-15-0184-5_29

[4] Li, M., Li, Y., Lou, W., Chen, L. (2020). A hybrid recommendation system for Q&A documents. Expert Systems with Applications, 144: 113088. https://doi.org/10.1016/j.eswa.2019.113088

[5] Hasanzadeh, N., Forghani, Y. (2019). Improving the accuracy of M-distance based nearest neighbor recommendation system by using ratings variance. Ingénierie des Systèmes d'Information, 24(2): 131-137. https://doi.org/10.18280/isi.240201

[6] Chen, L., Wu, Z., Cao, J., Zhu, G., Ge, Y. (2020). Travel recommendation via fusing multi-auxiliary information into matrix factorization. ACM Transactions on Intelligent Systems and Technology (TIST), 11(2): 1-24. https://xs.scihub.ltd/https://doi.org/10.1145/3372118

[7] Mansoury, M., Shajari, M. (2016). Improving recommender systems' performance on cold-start users and controversial items by a new similarity model. International Journal of Web Information Systems, 12(2): 126-149. https://doi.org/10.1108/IJWIS-07-2015-0024

[8] Akama, S., Kudo, Y., Murai, T. (2020). Neighbor selection for user-based collaborative filtering using covering-based rough sets. In Topics in Rough Set Theory, 168: 141-159. https://doi.org/10.1007/978-3-030-29566-0_9

[9] Mustaqeem, A., Anwar, S.M., Majid, M. (2020). A modular cluster based collaborative recommender system for cardiac patients. Artificial Intelligence in Medicine, 102: 101761. https://doi.org/10.1016/j.artmed.2019.101761

[10] Zheng, M., Min, F., Zhang, H.R., Chen, W.B. (2016). Fast recommendations with the m-distance. IEEE Access, 4: 1464-1468. https://doi.org/10.1109/ACCESS.2016.2549182

[11] Lemire, D., Maclachlan, A. (2005). Slope one predictors for online rating-based collaborative filtering. In Proceedings of the 2005 SIAM International Conference on Data Mining, pp. 471-475. https://doi.org/10.1137/1.9781611972757.43

[12] Wang, Q.X., Luo, X., Li, Y., Shi, X. Y., Gu, L., Shang, M.S. (2018). Incremental slope-one recommenders. Neurocomputing, 272: 606-618. https://doi.org/10.1016/j.neucom.2017.07.033

[13] Feng, L., Zhao, Q., Zhou, C. (2020). Improving performances of Top-N recommendations with co-clustering method. Expert Systems with Applications, 143: 113078. https://doi.org/10.1016/j.eswa.2019.113078

[14] Bu, J., Shen, X., Xu, B., Chen, C., He, X., Cai, D. (2016). Improving collaborative recommendation via user-item subgroups. IEEE Transactions on Knowledge and Data Engineering, 28(9): 2363-2375. https://doi.org/10.1109/TKDE.2016.2566622

[15] Khan, Z., Iltaf, N., Afzal, H., Abbas, H. (2020). Enriching non-negative matrix factorization with contextual embeddings for recommender systems. Neurocomputing, 380: 246-258. https://doi.org/10.1016/j.neucom.2019.09.080

[16] Wang, Y., Feng, C., Guo, C., Chu, Y., Hwang, J.N. (2019). Solving the sparsity problem in recommendations via cross-domain item embedding based on co-clustering. In Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, pp. 717-725.

NOMENCLATURE

T	User-item matrix which contain both rated and unrated items
E	Item tag
c	The number of groups
k	Each user or item is the member of how many groups
R	The dimension of spectral space
$\tilde{T}$	New User-item matrix with estimated rates of unrated items
Q	The membership degree matrix of users in groups
R	The membership degree matrix of items in groups
P	Join of Q and R