I.J. Education and Management Engineering, **2020, 1, 27-42**
Published Online February 2020 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijeme.2020.01.04

Available online at ***http://www.mecs-press.net/ijem***

An Enhanced Data Sparsity Reduction Method for Effective Collaborative Filtering Recommendations

[1]Abubakar Roko*, [2]Abba Almu, [3]Aminu Mohammed and [4]Ibrahim Saidu

[1, 2, 3] Dept. of Mathematics, Computer Science Unit, Usmanu Danfodiyo University, P.M.B 2346, Sokoto – Nigeria
[4]Dept. of Information and Communications Technology, Usmanu Danfodiyo University, P.M.B 2346, Sokoto – Nigeria

Received: 29 September 2019; Accepted: 26 October 2019; Published: 08 February 2020

Abstract

Collaborative filtering recommender system suffers from data sparsity problem due to its reliance on numerical ratings to provide recommendations to users. This problem makes it difficult for the system to compute accurate similar neighbours for the items and provide good quality recommendations. Existing methods fail to pre-process the missing ratings of the new items and to predict cold items to the active users which lead to poor quality recommendations. In this work, a sparsity reduction method is presented to improve the quality of recommendations. The method utilises Bi-Separated clustering algorithm to cluster the ratings matrix simultaneously into users and items bi-clusters based on ratings classification. It also employs Bi-Mean Imputation algorithm to fill the missing ratings in the bi-clusters using the estimated means. The method then performs the traditional collaborative filtering process on the new rating matrix for cold items prediction. The experimental results demonstrated that compared to the existing method, the proposed BiSCBiMI improves density of the rating matrix by 5.75%, 10.73% and 7.35% as well as Mean Absolute Error (MAE) of the new items prediction for all of the considered datasets. The results indicated that, the proposed approaches are effective in reducing the data sparsity problem as well as items prediction, which in turn returns good quality recommendations.

Index Terms: collaborative filtering, data sparsity, bi-separated clustering, bi-mean imputation, cold item, rating matrix

© 2020 Published by MECS Publisher. Selection and/or peer review under responsibility of the Research Association of Mode rn Education and Computer Science

* Corresponding author.
E-mail address:

1. Introduction

The huge amount of information on the web and the emergence of e-commerce have led to the problem of information overload. Because of this problem, it becomes difficult for users to search for items of their interest. Therefore, a recommender system is essential in order to identify items based on user's interest. The system is an information filtering system that recommends relevant items to users by analyzing the users explicitly mentioned preferences and interests [21, 11]. It saves a lot of time and effort of users typically involve in issuing different queries about the items of interest, by simply prioritising and personalising large volume of information available at its disposal to find the unknown relevant items needed by the users. This prompted many research efforts on recommender systems [8, 2, 14, 16, 18, 3]. Among these systems include Collaborative Filtering (CF) which is the most popular and successful system that provides recommendations to users because it recommends any type of items to users such as books, movies, news, music, web pages and so forth [5, 19].

A typical CF recommender system requires some representations of users and items. This system generates an individual's interest based on other similar users. The general idea is, given a user u, the system ranks other users based on similarity with users, u_1,…,u_m. It then predicts user preferences based on the ratings of 1, 2,…, m users. The rating is determined according to a common set of objects or items o_1,…,o_n. Normally, the users and objects are arranged in a matrix called user-item rating matrix. The matrix is processed using algorithms to generate the recommendations. These algorithms first use similarity functions to select users (or items) that are similar to the active user, which is the person that needs the recommendations.

However, most of the users ignore to rate most items in the user-item rating matrix or some items have few available ratings [13], which leads to data sparsity problem [26]. Therefore, it makes the generation of accurate neighbourhood for users or items impossible to predict correct item rating based on user's preference. This affects the accuracy of items prediction which in turn results in poor quality recommendations to users.

Numerous studies have been conducted on CF systems to deal with the data sparsity problem for predicting correct item rating in order to provide good recommendations [15, 13, 23, 25, 5,7, 20, 22, 17, 1, 12, 4]. Despite these studies, several challenges are left on resolved, which include: The existing method [4] fails to pre-process the missing ratings of the new items which increase the sparsity of the rating matrix. It also ignores to predict cold items to the active users which lead to poor quality recommendations.

In this work, a sparsity reduction method is presented to improve the quality of recommendations. The method uses Bi-separated clustering and Bi-Mean imputation algorithms to obtain new rating matrix with reduced sparsity of the rating matrix. The method then applies the CF process on the new rating matrix to predict cold items. The results of the evaluation indicated that, the proposed method improves density and MAE of the new items prediction.

2. Related Works

In this section, a review of several existing approaches, methods and algorithms proposed to alleviate the effect of data sparsity problem in CF recommender systems are presented.

Ma, King and Lyu [15] presented an effective algorithm for missing data prediction in collaborative filtering. The algorithm utilises the information from users or items correlations (or both) to predict missing value in the user-item rating matrix that can only help in providing positive recommendations to the active users. The new algorithm is identified to be more robust in dealing with data sparsity than the existing user-based or item-based algorithms. However, it suffers from data sparsity problem when there are fewer ratings for the active users which may lead to inaccurate prediction.

Liang, Bo and Jun [13] proposed a hybrid approach to overcome data sparsity in CF systems. The approach uses a similarity weight to deal with the significant influence of data sparsity by unifying the user-based and item-based approaches to provide accurate rating prediction to the active users. The new approach improves prediction accuracy while at the same time reducing the sparsity problem of the rating matrix. However, the approach finds it impossible to make prediction when the sparsity of the rating matrix is very high.

Xia et al. [23] presented strategies to alleviate data sparsity of the user-item rating matrix based on missing data imputation. The strategies utilise some user's demographic data consisting of age range and occupation to improve CF process. The demographic data is used together with the original ratings to fill up the user-item rating matrix and then perform the CF on the modified ratings. The strategies incorporation proves to be effective on the recommendation accuracy even under high sparse data. However, the proposed data imputation method fails to select the real similar neighbours of the items by considering the ratings of users within the same age or occupation to have close ratings on the same items.

Tan and Ye [25] developed an algorithm based on item classification to fill the vacant ratings of the rating matrix. The algorithm clusters the items in some classification to pre-produce ratings for the missing values in the matrix. The new rating matrix generated is used to provide recommendations. The algorithm alleviates the sparsity problem of the rating matrix while at the same provides better recommendations than the existing CF algorithms. However, the algorithm may provide irrelevant recommendations in a situation where the user's interest differs on the items within the same classification.

Chen et al. [5] introduced an algorithm to solve sparsity problem by using associative retrieval approach. The algorithm tries to capture the user's interest utilising both direct and indirect similarity between users ratings relative distance to compute similarity matrix. It uses associative retrieval to look for transitive associations on the users' ratings dataset to ease the prediction. The associative retrieval incorporation resulted in reducing the sparsity of the rating matrix thereby producing recommendations with good coverage and quality. However, the algorithm relies on the single source of information from the ratings to deal with the sparsity problem.

Chujai et al. [7] proposed an approach for imputing missing values based on the pattern of frequent itemsets to address sparsity problem. The approach utilises not only the ratings for the items but include more demographic information of the user and item to fill up the missing rating data in the user-item rating matrix by mining frequent itemsets in the dataset. It demonstrated the ability to fill up the missing ratings with improved recommendations accuracy. However, the approach fails to reduce the sparsity of the rating matrix if it contains many infrequent itemsets.

Qin, Cao and Peng [20] presented an algorithm to solve missing value in CF recommendation process. The algorithm considers the user's subjective preferences such as scoring habits and item's objective quality to compute the value of unrated items whether frequent or infrequent in the rating matrix. The computed value is used to replace the unrated item value in the matrix. It improves the recommendation quality of the sparse rating matrix. However, the algorithm may find it difficult to ensure quality recommendations when the dimension of the rating matrix is very high.

Song [22] introduced a collaborative filtering algorithm to alleviate data sparsity problem based on Multi-Dimensional Data Filing (MDDF). The algorithm chooses dimensions that can affect users' preferences on the items and then the missing values in the user-item rating matrix are filled with multidimensional data of the chosen dimensions. It reduces the degree of the sparsity of the rating matrix and the efficiency of the users' recommendations is increased. However, the algorithm utilises additional information such as users' age and gender which may be insufficient, thereby causing extra burden during items selection with missing ratings to be filled.

Najafabadi et al. [17] proposed a technique based on clustering and association rules mining to solve data sparsity problem without using users' associated features. The technique utilises clustering method to reduce the size of the sparse data in the item space and also employs association rules mining method to detect

similar patterns in user-item interactions. It achieves better recommendations performance than the existing CF methods compared, even when the sparsity of the dataset is high. However, the technique used implicit feedback to process prediction, which might be unable to produce accurate prediction on the explicit ratings.

Ardimansyah, Huda and Baizal [1] presented an approach based on matrix factorisation pre-processing to solve the problem of sparse explicit rating data for memory-based CF recommender. The approach pre-processes empty rating values by filling them with the scores estimated from matrix factorisation. It enhances the performance of CF-based methods with pre-processing by providing better prediction accuracy than those without pre-processing. However, the estimated rating scores might be unable to reasonably express users' preferences on the different category of items.

Li et al. [12] introduced Category Preferred Canopy-K-means based Collaborative Filtering Algorithm (CPCKCF) to alleviate the sparsity and scalability of data challenges in recommender systems. The CPCKCF reduces the dimension of the sparse data by calculating the user-item category preferred ratio and then use it to cluster the users' data accordingly. The CPCKCF also improves the recommendation accuracy and efficiency of computation when compared to user-based CF algorithm. However, it ignores the rating diversity considering only one dimension when clustering items or users because some useful information in the opposite dimension might not be considered.

Cheng and Zhang [4] proposed a Jaccard Coefficient-based Bi-clustering and Fusion (JC-BiFu) method to alleviate the data sparsity problem in recommender system. The JC-BiFu utilises a density peak method to cluster both users and items in the user-item rating matrix. It then select the most similar cluster for both target user and item to estimate the missing values for unrated entries in the rating matrix. The prediction is done for the target user based on the similar neighbours. JC-BiFu method provides more accurate predictions with less error compared to other methods. However, the method ignores to pre-process new items with missing ratings in order to decrease the sparseness of the rating matrix. It also fails to provide rating prediction for the cold items. Consequently, the method returns poor quality recommendations to users.

3. Proposed Method

In this section, an enhanced sparsity reduction method based on Bi-Separated Clustering and Bi-Mean Imputation (BiSCBiMI) is proposed to reduce the sparseness of the rating matrix as well as to provide accurate prediction for the cold items in order to address the shortcomings of the JC-BiFu method identified earlier. The general framework for the BiSCBiMI method is presented in Fig. 1.

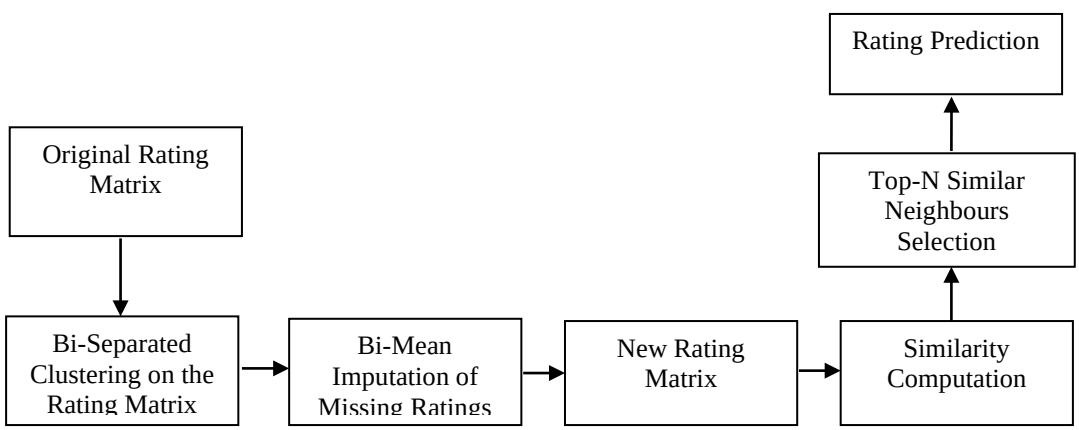


Fig. 1. Sparsity Reduction Framework of the Enhanced Method

A. Clustering Phase

First, JC-BiFu uses density peak clustering to cluster users and items by employing distance matrix to partition clusters based on cosine similarity, minimum points and cut-off distance. Only the data points within high density region that are density reachable and connected are assigned to a cluster.

TABLE 1. USER-ITEM RATING MATRIX

	U_1	U_2	U_3	U_4	U_5	U_6	U_7
I_1		5	2		4		3
I_2		4		3	2		4
I_3	3			3	5		2
I_4	4		5	3	1		1
I_5							
I_6	4	3	4	2	4		3

For instance, the JC-BiFu utilises the rating matrix in Table 1 and use the cosine similarity in Equation (1) to construct the distance matrices in Table 2 for partitioning the items into clusters.

$$Cos(i,j) = \frac{i \cdot j}{|i| \, x \, |j|} \tag{1}$$

Where the . represents dot product of the two vector i and j, $|i|$ indicates the length of the vector i and $|j|$ indicates the length of the vector j.

TABLE 2. ITEMS DISTANCE MATRIX

	I_1	I_2	I_3	I_4	I_5	I_6
I_1	0.0000	0.8115	0.5161	0.3208	0.0000	0.7807
I_2	0.8115	0.0000	**0.5871**	0.3101	0.0000	0.6770
I_3	0.5161	0.5871	0.0000	**0.5664**	0.0000	0.7670
I_4	**0.3208**	**0.3101**	0.5664	0.0000	0.0000	0.8121
I_5	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
I_6	0.7807	0.6770	0.7670	0.8121	0.0000	0.0000

From Table 2 only two clusters (I_1, I_2 and I_3, I_4) are formed based on the density connected and density reachable nearest points as highlighted in the dark boxes. To estimate the missing values in the rating matrix, JC-BiFu apply the Jaccard Coefficient to measure the similarity between items in clusters i and j respectively as defined in Equation (2).

$$J(i,j) = \frac{|U_i \cap U_j|}{|U_i \cup U_j|} \tag{2}$$

Where U_i is the set of users that rated item i, the numerator is the number of users that rated both item i and j, and the denominator is the number of users that rated item i or j.

The computed similarity is then used to estimate the missing values in Table 1 using the estimation method defined in Equation (3).

$$\hat{r}_{u,i} = \bar{r}_u + \frac{\sum_{i,j \in I_c} J(i,j)(r_{u,j} - \bar{r}_u)}{\alpha + \sum_{i,j \in I_c} J(i,j)} \quad (3)$$

Where $\hat{r}_{u,i}$ is the rating estimate for user u on item i, $\bar{r}_u$ is the user u average rating, I_c is the item cluster consisting of items i and j, and $\alpha > 0$ is a parameter to smooth the estimation for avoiding denominator to be 0.

The application of Equation (3) resulted in the rating matrix with less missing ratings issues as depicted in Table 3.

TABLE 3. JC-BIFU RESULTANT RATING MATRIX

	U_1	U_2	U_3	U_4	U_5	U_6	U_7
I_1	5	5	2	5	4		3
I_2	5	4	5	3	2		4
I_3	3	4	4	3	5		2
I_4	4	4	5	3	1		1
I_5							
I_6	4	3	4	2	4		3

Therefore, JC-BiFu reduces the sparsity problem in the rating matrix. However, it is unable to estimate the ratings for the new user (U6) or new item (I5) as shown in Table 3 which in turn increases the sparseness of the rating matrix.

Second, in order to alleviate the sparsity problem beyond JC-BiFu, the proposed BiSCBiMI method utilises bi-separated clustering technique to partition the ratings matrix into users and items bi-clusters (UC and IC) respectively. The ratings are assigned to the nearest cluster neighbour based on high rating (3, 4 or 5) or low rating (1 or 2) classes since each rating represents an integer value from 1 to 5. The following two assumptions were made for clustering with the presence of missing ratings:

(i) The maximum empty separation points denoted by ε within each cluster is less than or equal to two i.e. $\varepsilon \in \{0, 1, 2\}$.

(ii) The minimum non-empty points denoted by minNpts within each cluster is greater than or equal to two i.e. $\text{minNpts} \in \{2, 3, 4, \ldots\}$.

Hence, the Equation (4) and Equation (5) defined the bi-clusters. The Algorithm 1 demonstrated the procedure involved during partitioning the users and items into clusters.

$$IC_n = \{ic_1, ic_2, ic_3, \ldots, ic_n\} \quad (4)$$

$$UC_m = \{uc_1, uc_2, uc_3, \ldots, uc_m\} \quad (5)$$

Where IC represents item cluster, UC represents user cluster, ic_n indicates item sub cluster n and uc_m indicates user sub cluster m.

Algorithm 1: Bi-Separated Clustering Algorithm

Input: dataset - - containing user–item ratings
Output: *IC*, *UC* clusters

1. Given two parameters: ε and minNpts with high and low rating specification
2. while ε ≤ 2 and minNpts ≥ 2 do
3. form k random clusters for both users and items
4. for all items partition the ratings into clusters using Equation (4)
5. if the rating >= high then
6. assign it to a high item sub-cluster
7. else
8. assign it to a low item sub-cluster
9. for all users partition the ratings into clusters using Equation (5)
10. if the rating >= high then
11. assign it to a high user sub-cluster
12. else
13. assign it to a low user sub-cluster
14. repeat step 4-8 and step 9-13 until no more clusters can be form
15. return items and user clusters

Now, the proposed Bi-Separated Clustering Algorithm is applied on the user-item rating matrix of Table 1 to cluster users and items simultaneously. The Algorithm produces the result in Table 4 with different clusters.

TABLE 4. BI-SEPARATED CLUSTERING OF THE RATING MATRIX

	U_1	U_2	U_3	U_4	U_5	U_6	U_7
I_1		5	2		4		3
I_2		4		3	2		4
I_3	3			3	5		2
I_4	4		5	3	1		1
I_5							
I_6	4	3	4	2	4		3

Having clustered the rating matrix, it is observed that, for any two or more non-empty rating points ($x_1, x_2, \ldots x_k$) from the same cluster say ic_n or uc_m, a solution β exists that belongs to ic_n or uc_m as indicated in Equation (6).

$$\frac{x_1 + x_2 + \ldots x_k}{k} = \beta \in \quad ic_n \text{ or } uc_m \tag{6}$$

Since β exists in ic_n or uc_m cluster, the bi-mean imputation technique is employed to estimate the missing ratings in the bi-clusters. The technique performs simultaneous filling of the missing ratings in the item or user bi-clusters by using of mean rating computation for each of the users or items cluster as in equation (7) and equation (8).

$$ic\_mean = \frac{\sum_{i=1}^{n} rIC_{ki}}{n} \tag{7}$$

Where, ic_mean stands for the mean of item cluster, rIC_{ki} represents the rating of the k[th] cluster for items i

and k is constant.

$$uc\_mean = \frac{\sum_{j=1}^{m} rUC_{pj}}{m} \quad (8)$$

Where, uc_mean stands for the mean of user cluster, rUC_{pi} represents the rating of the p[th] cluster for users j and p is constant.

The unknown ratings of the users or items in the partitioned clusters are imputed using the calculated mean ratings of all other users or items belonging to the cluster as demonstrated by Algorithm 2. Then the resultant rating matrix with imputed values is obtained to form the so called new rating matrix suitable for CF process.

Algorithm 2: Bi-Mean Imputation Algorithm

Input: A rating matrix - - containing items and users clusters = {IC, UC}
Output: new rating matrix
1. for each IC clusters
2. compute the ic_mean of each item cluster whether high or low using (7)
3. if the rating in each item cluster is null then
4. replace it with the calculated cluster mean
5. for each UC clusters
6. compute the uc_mean of each user cluster whether high or low using (8)
7. if the rating in each user cluster is null then
8. replace it with the calculated cluster mean
9. repeat step 1-4 and step 5-8 until no more missing ratings in the clusters
10. merge IC and UC clusters
11. return new rating matrix

Table 5 presented the Bi-Mean Imputation Algorithm simultaneous computation on the users and items clusters of Table 4. The Algorithm then merges the clusters into a single cluster known as new rating matrix.

TABLE 5. BI-MEAN IMPUTATION RESULT

	U_1	U_2	U_3	U_4	U_5	U_6	U_7
I_1	5	5	2		4	4	3
I_2	5	4	3	3	2		4
I_3	3		3	3	5		2
I_4	4		5	3	1	1	1
I_5	4		5				
I_6	4	3	4	2	4	4	3

As seen in Table 5 unlike JC-BiFu method on Table 3, the proposed BiSCBiMI method is able to pre-process and estimate the missing ratings for the new users and the new items. This reduces the sparsity of the rating matrix where most data cells are none empty than when JC-BiFu method is employed.

B. Prediction Phase

At this phase, the prediction can be computed for the active user u on the target item i if the set of similar neighbours of that user are obtained. To obtain similar neighbours for both JC-BiFu and BiSCBiMI methods respectively in Table 3 and Table 5 for the cold items rating prediction.

First, the modified cosine similarity function in Equation (9) is used to measure the similarity between item i and the remaining item j.

$$\mathbf{sim(i,j)}_{=} \frac{\sum_{d\in U}(R_{di}-\overline{R}_i)(R_{dj}-\overline{R}_j)}{\sqrt{\sum_{d\in U}(R_{di}-\overline{R}_i)^2}\sqrt{\sum_{d\in U}(R_{dj}-\overline{R}_j)^2}} \quad (9)$$

Where, R_{di} means the rating of user *d* on item *i* and R_i the average rating of all items rated by the user d. *U* is the set of users that rated both items *i* and *j*.

Second, having calculated the set of item neighbours the prediction is done using item-based CF to predict the rating for the cold items as in Equation (10).

$$P_{di} = \overline{R}_{\mathrm{d}} + \frac{\sum_{e=1}^{n}(R_{ei}-\overline{R}_e)\ \mathrm{x}\ \mathrm{sim(i,j)}}{\sum_{e=1}^{n}|sim(i,j)|} \quad (10)$$

Where, P_{di} is the predicted rating for cold item *i* to user *d*, $\overline{R}_{\mathrm{d}}$ is the user *d* average rating, *n* is the total number of top similar neighbours selected and R_{ei} is the rating for user *e* on item *i*.

TABLE 6. COLD ITEM PREDICTION

Item Predicted	Similarity of Neighbours		Prediction Value	
	JC-BiFu	BiSCBiMI	JC-BiFu	BiSCBiMI
$P_{u2,I5}$	0.0000,0.0000	0.4676, 0.7990	-	4.11
$P_{u4,I5}$	0.0000,0.0000	0.4676, 0.7990	-	2.86
$P_{u6,I2}$	0.0000,0.0000	-0.1376, -0.1771	-	2.69

Table 6 describes the cold item prediction from Table 3 and Table 5 for both JC-BiFu and BiSCBiMI methods. It is clearly seen that, JC-BiFu is unable to find any similar neighbours for the cold item (I5) and hence cannot predict this item to any of the users. Whereas the BiSCBiMI find some similar neighbours for the cold item (I5) and is able to predict this item to user 2 (u2) and user 4 (u4) respectively. The method is also capable of predicting item 2 (I2) to the brand new user 6 (u6) who have not rated any item initially.

4. Experimental Evaluation and Results

This section describes the datasets, experimental environment and evaluation metrics used to assess the effectiveness of the proposed BiSCBiMI method as well as the presentation and the description of the results obtained.

A. Description of Dataset

The three well known datasets are used to evaluate the performance of the proposed BiSCBiMI method compared to JC-BiFu. These datasets consist of FilmTrust, MovieLens-100K and MovieLens-1M [9] with detail description in Table 7 before the experiments. The FilmTrust dataset is obtained from librec site while MovieLens-100K and MovieLens-1M are obtained from the widely used MovieLens recommender site developed by the GroupLens research team at the University of Minnesota, United State. These datasets are real-world datasets used by different researchers to test many of the core algorithmic advances in recommender system researches [10].

TABLE 7. DATASETS INFORMATION

Datasets	Number of Users	Number of Items	Number of Ratings	Sparsity (%)	Density (%)
FilmTrust	1, 508	2, 071	35, 497	98.86	1.14
MovieLens-100K	943	1, 682	100, 000	93.70	6.30
MovieLens-1M	6, 040	3, 952	1, 000, 209	95.81	4.19

B. Experimental Environment

The experiment is conducted on the computer system with the following specifications: Windows 8, 64-bit operating system, Intel (R) Core ((TM) i3-3120M CPU @ 2.5GHz) and the memory size of 4.0GB. The tools used for the experimental programs are PyScripter Integrated Development Environment (IDE) and Python programming language.

C. Evaluation Metrics

The density and accuracy of the proposed BiSCBiMI method is evaluated using the following performance metrics:

(i) Density is a metric used to measure the percentage of the populated ratings on the rating matrix and it is number of total rating entries divide by the number of users multiplied by the number of items [24]. It is computed as shown in Equation (11).

$$Density = \left(\frac{\#Tratings}{\#UsersX\#Items}\right) X\ 100 \tag{11}$$

Where, $\#Tratings$ is the total number of ratings, $\#Users$ is the number of users and $\#Items$ is the number of items. The higher density value the lesser sparsity value of the user-item rating matrix.

(ii) Sparsity Level is a metric used to measure the percentage of missing ratings on the rating matrix and it is defined as one minus the density as shown in Equation (12).

$$Sparsity\ Level = 100 - Density \tag{12}$$

(iii) **Mean Absolute Error (MAE)** is an accuracy metric used to find the deviations between the predicted rating scores of the proposed system against the user actual rating scores for the items in collaborative filtering recommendations [6]. This metric is defined in Equation (13).

$$MAE = \frac{\sum_{i=1}^{N} |r_{u,i} - \hat{r}_{u,i})|}{N} \tag{13}$$

Where, N is the number of corresponding ratings-prediction pairs $r_{u,i}$is the actual item rating and $\hat{r}_{u,i}$ is the predicted item rating. The lower the MAE value obtained the higher the accuracy of the prediction.

D. Experimental Results

This section reports the results of the experiments of the proposed BiSCBiMI method compared to JC-BiFu method based on three datasets described in Table 7 in terms of density and MAE metrics.

1) Density Results Analysis

The effect of the JC-BiFu and BiSCBiMI methods on three datasets is evaluated using the density metric in order to show the sparsity level of the rating matrix. The evaluation results are presented in Tables 8, 9 and 10 respectively.

TABLE 8. DENSITY ON FILMTRUST DATASET

Method	Density (%)	Sparsity Level (%)
JC-BiFu	6.83	93.17
BiSCBiMI	12.58	87.42

TABLE 9. DENSITY ON MOVIELENS-100K DATASET

Method	Density (%)	Sparsity Level (%)
JC-BiFu	13.43	86.57
BiSCBiMI	24.16	75.84

TABLE 10. DENSITY ON MOVIELENS-1M DATASET

Method	Density (%)	Sparsity Level (%)
JC-BiFu	10.39	89.61
BiSCBiMI	17.74	82.26

Table 8 illustrates the density of the two methods on the FilmTrust dataset. Both the densities of the methods increased thereby decreasing the sparsity level of the dataset. Thus, the density of the BiSCBiMI is higher than that of JC-BiFu by 5.75%. This difference occurs as a result of the ability of BiSCBiMI to estimate the missing ratings with the inclusion of both new items and users which is not obtainable in JC-BiFu. It is clear that, BiSCBiMI outperforms JC-BiFu method in reducing the sparsity on FilmTrust dataset.

Tables 9 and 10 show the influence of the two methods on the MovieLens-100K and MovieLens-1M datasets using density metric. The densities of the two methods increased on both datasets thereby indicating decrease in the sparsity levels. The proposed BiSCBiMI method outperforms JC-BiFu method with higher density on both MovieLens-100K and MovieLens-1M datasets by 10.73% and 7.35% respectively. This implies that, BiSCBiMI method demonstrated a better performance in sparsity reduction by estimating the missing ratings for both new and existing items based on the sparsity levels values obtained on both datasets

as well as higher rating density. Thus, the BiSCBiMI method can be effective for recommender system especially in sparse user-item rating matrices.

2) MAE Results Analysis

The accuracy of proposed BiSCBiMI method and existing JC-BiFu method is tested on three datasets in order to verify their prediction performance. The evaluation results are presented in Figures 3, 4 and 5 respectively.

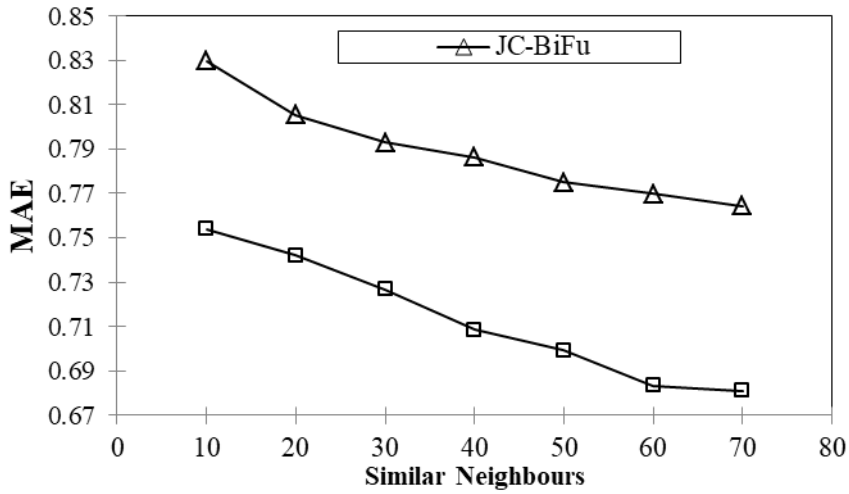


FIG. 2. MAE ON FILMTRUST DATASET

It is shown on Fig. 2 that when the number of similar neighbours is small both the two methods have low prediction performance on FilmTrust dataset. But, when the number of similar neighbours is greater than or equal to 50, the BiSCBiMI method has better prediction performance than the JC-BiFu method because of its lowest MAE value of 0.6811 against 0.7643.

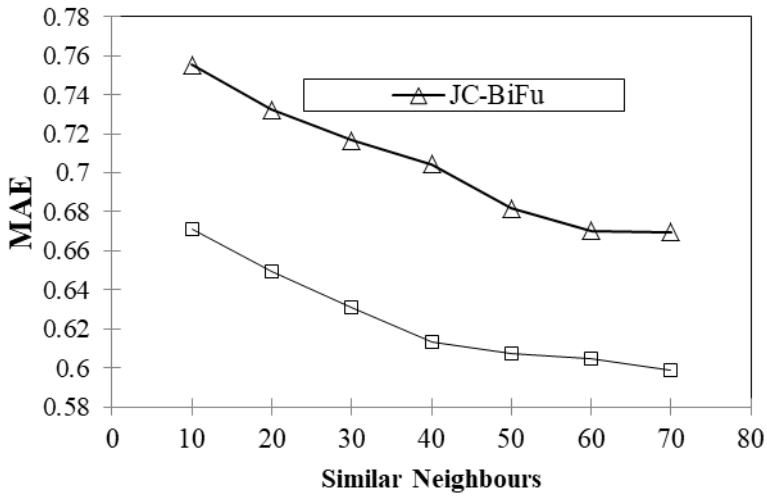


FIG. 3. MAE ON MOVIELENS-100K DATASET

Fig. 3 shows that, when the number of similar neighbours is small both the two methods have low prediction performance on MovieLens-100K dataset. But, when the number of similar neighbours is greater than or equal to 50, the BiSCBiMI method has better prediction performance than the JC-BiFu method due to its the lowest MAE value of 0.5986 against 0.6690.

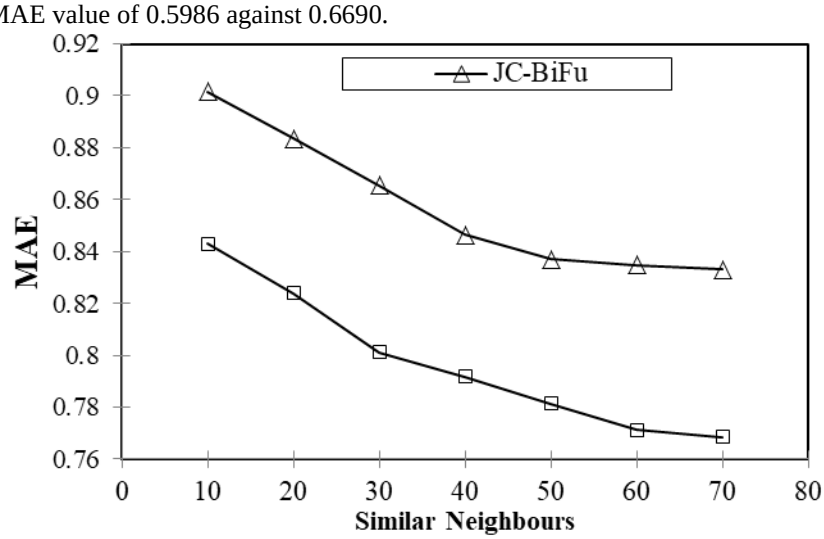


FIG. 4. MAE ON MOVIELENS-1M DATASET

Fig. 4 illustrates the prediction accuracy of the two methods on MovieLens-1M dataset. Similar to Figure 3, when the number of similar neighbours is small a low prediction performance is noticed. But, when the number of similar neighbours is greater than or equal to 50, the BiSCBiMI method demonstrated a better

prediction performance than the JC-BiFu method because of its lowest MAE value of 0.7685 against 0.8331.

It can be seen from the above results that, the BiSCBiMI method achieved the lowest MAE values than JC-BiFu for both small number of neighbours and large number of neighbours on all the three datasets. Also, the BiSCBiMI method achieved the lowest MAE on MovieLens-100K dataset than the remaining two datasets. Therefore, it demonstrated the best prediction performance which results in quality recommendations to the users. It also shows the ability for identifying similar neighbours for the cold items and then provides prediction for those items appropriately.

5. Conclusion

In this research work, a BiSCBiMI is proposed to automatically pre-process the missing ratings of the new and existing items of the rating matrix so as to improve the quality of the recommendations. Specifically, the proposed method employs BiSC algorithm to cluster the ratings matrix simultaneously into users and items bi-clusters by utilising ratings classification. The method then engages BiMI algorithm to estimate the missing ratings in each cluster using the calculated means. Next, the method applies traditional CF to the new rating matrix for cold items prediction. The experimental results demonstrated that, the proposed BiSCBiMI method outperforms JC-BiFu method in terms of sparsity reduction level and lower prediction error on all the three datasets. However, as a future work, the effect of implicit datasets on the proposed BiSCBiMI method will be experimentally investigated.

References

[1] Ardimansyah, I. M., Huda, F. A., and Baizal, A. Z. K.,. Preprocessing Matrix Factorization for Solving Data Sparsity on Memory-based Collaborative Filtering. In Proceedings of the 3rd International Conference on Science in Information Technology (ICSITech), 2017, pp. 521-525, Bandung, Indonesia.

[2] Burke, R. Hybrid Recommender Systems. User Modeling and User-Adapted Interaction, 2002, 12(4), 331-370.

[3] Cacheda, F., Carneiro, V., Fernandez, D., and Formoso, V. Comparison of Collaborative Filtering Algorithms: Limitations of Current Techniques and Proposal for Scalable, High-performance Recommender Systems. ACM Transactions on the Web, 2011, 5(1), 1-33.

[4] Cheng J., Zhang L. Jaccard Coefficient-Based Bi-clustering and Fusion Recommender System for Solving Data Sparsity. Advances in Knowledge Discovery and Data Mining, 2009, 1144: 369-380, Springer, Cham.

[5] Chen, Y., Wu, C., Xie, M., and Guo, X. Solving the Sparsity Problem in Recommender Systems Using Association Retrieval. Journal of Computers, 2011, 6(9):1896-1902.

[6] Chen, Z., Jiang, Y., and Zhao, Y. A Collaborative Filtering Recommendation Algorithm Based on User Interest Change and Trust Evaluation. International Journal of Digital Content Technology and its Applications, 2010, 4(9), 106-113.

[7] Chujai, P., Rasmequan, S., Suksawatchon, U., and Suksawatchon, J. Imputing Missing Values in Collaborative Filtering Using Pattern Frequent Itemsets. In Proceedings of the International Electrical Engineering Congress (iEECON), 2014, pp. 694-697, Chonburi, Thailand.

[8] Goldberg, D., Nichols, D., Oki, B., M., and Douglas, T. Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM, 1992, 35(12), 61-70.

[9] Guo, G., Zhang, J., and Yorke-Smith, N. A Novel Bayesian Similarity Measure for Recommender Systems. In Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI), 2013, pp. 2619-2625, Beijing, China.

[10] Harper, F., M., and Konstan, J., A. The MovieLens Datasets: History and Context. ACM Transactions on Interactive Intelligent Systems (TiiS), 2015, 5(4), 1-19.

[11]Khusro, S., Ali, Z., and Ullah, I. Recommender Systems: Issues, Challenges and Research Opportunities. Lecture Notes in Electrical Engineering, 2016, 376, 1179-1189. Springer, Science+Business Media Singapore.

[12] Li, J., Zhang, K., Yang, X., Wei, P., Wang, J., Mitra, K., and Ranjan, R., Category Preferred Canopy–K-Means based Collaborative Filtering Algorithm. Future Generation Computer Systems, 2018, 93: 1046-1054.

[13] Liang, Z., Bo, X., and Jun, G. A Hybrid Approach to Collaborative Filtering for Overcoming Data Sparsity. In Proceedings of the 9th IEEE International Conference on Signal Processing, 2008, pp. 1596-1599, Beijing, China.

[14] Linden, G., Smith, B., and York, J. Amazon.com Recommendations: Item-to-Item Collaborative Filtering. IEEE Computer Society, 2003, pp. 76-80.

[15] Ma, H., King, I., and Lyu, R. M. Effective Missing Data Prediction for Collaborative Filtering. In Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2007, pp. 39-46, Amsterdam, Netherlands.

[16] Miller, B., N., Albert, I., Lam, S., K., Konstan, J., A., and Riedl, J. MovieLens Unplugged: Experiences with an Occasionally Connected Recommender System. Proceedings of the 8th International Conference on Intelligent User Interfaces (IUI), 2003, pp. 263-266, Miami, Florida, USA.

[17] Najafabadi, K. M., Mahrin, N. M., Chuprat, S., and Sarkan, H. M. Improving the Accuracy of Collaborative Filtering Recommendations Using Clustering and Association Rules Mining on Implicit Data. Computers in Human Behavior, 2017, 67: 113-128.

[18] Pazzani, M., J., and Billsus, D. Content-Based Recommendation Systems. The Adaptive Web-Springer, 2007, 4321: 325-341.

[19] Ping, H. Q., and Ming, X. Research on Several Recommendation Algorithms. Procedia Engineering, 2012, 29: 2427-2431.

[20] Qin, J., Cao, L., and Peng, H. A Solution of Missing Value in Collaborative Filtering Recommendation Algorithm. In Proceedings of the IEEE Chinese Automation Congress (CAC), 2015, pp. 2184-2187, Wuhan, China.

[21] Resnick, P., and Varian, H. Recommender Systems. Communications of the ACM, 1997, 40(3):56-58.

[22] Song, M. A Collaborative Filtering Recommendation Algorithm Based on Multi-dimensional Data Filling. In Proceedings of the 2nd IEEE International Conference on Computer and Communications (ICCC), 2016, pp. 175-179, Chengdu, China.

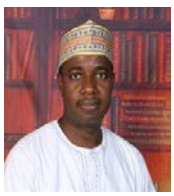
[23] Tan, H., and Ye, H. A Collaborative Filtering Recommendation Algorithm Based on Item Classification. In Proceedings of the IEEE Pacific-Asia Conference on Circuits, Communications and Systems, 2009, pp. 694-697, Chengdu, China.

[24] Wang, J., Song, H., and Zhou, X. A Collaborative Filtering Recommendation Algorithm Based on Biclustering. In Proceedings of the International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2015, pp. 803-807, Shenyang, China.

[25] Xia, W., He, L., Gu, J., and He, K. Effective Collaborative Filtering Approaches Based on Missing Data Imputation. In Proceedings of the Fifth IEEE International Joint Conference on INC, IMS and IDC, 2009, pp. 534-537, Seoul, South Korea.

[26] Yongsheng, H., Xiangwu, M., and Yujie, Z. A Collaborative Filtering Recommendation Method to the Loyal-user Problem. In Proceedings of 2nd IEEE International Conference on Computer Science and Information Technology, 2009, pp. 57-60, Beijing, China.

Author's Profile



Abubakar Roko is a Senior lecturer at the Computer Science Unit, Faculty of Science, Usmanu Danfodiyo Univesity Sokoto, Nigeria. He received his PhD from Univeristy Putra Malaysia in 2016, specializing in the field of XML Retrieval. Currently, his research interest is in the area of text data management and analysis, focusing in particular on query processing in Information retrieval, Recommender Systems, and Sentiment Analysis.

Abba Almu is currently a PhD student at the Department of Mathematics, Computer Science Unit, Usmanu Danfodiyo University, Sokoto. His research interests include recommender systems and machine learning.

Aminu Mohammed is a professor at the Department of Mathematics, Computer Science, Unit of Usmanu Danfodiyo University, Sokoto-Nigeria. His current research interests include performance modelling and evaluation of wired/wireless networks protocols, high-performance networks, and distributed systems.

Ibrahim Saidu received the B.Sc. and M.Sc. degrees in Mathematics from Usmanu Danfodiyo University Sokoto, Nigeria and Bayero University Kano, Nigeria, respectively. He also received the Bachelor in Information Technology from Almadinah International University. Malaysia. Postgraduate Diploma in Computer Science from Federal University Technology. Minna. Nigeria and the M.Sc. in Computer Science with specialisation in Distributed Computing from Universiti Putra Malaysia (UPM). In addition, he received the Ph.D. in Computer Networks at UPM. His research interests include Performance Evaluation, Resource Management in Wireless Networks.

How to cite this paper: Abubakar Roko, Abba Almu, Aminu Mohammed, Ibrahim Saidu, " An Enhanced Data Sparsity Reduction Method for Effective Collaborative Filtering Recommendations ", International Journal of Education and Management Engineering(IJEME), Vol.10, No.1, pp.27-42, 2020.DOI: 10.5815/ijeme.2020.01.04