Transactions on Machine Intelligence 5(2) (2022) 97–114

ISSN Online: 2821-1693

Transactions on Machine Intelligence

Journal Homepage: https://tmachineintelligence.ir/index.php/TMI

Improved Recommender Systems Using Data Mining

H. Avini[1,*], Z. Mirzaei ZavardJani[2], A. Avini[3]

[1] Department of Computer Engineering, Faculty of Technology and Engineering, Yasouj Branch, Islamic Azad University, Yasouj, Iran
[2] Department of Computer Engineering, Faculty of Technology and Engineering, Qeshm Branch International University, Qeshm, Iran
[3] Department of Accounting, Faculty of Accounting, Yasouj Branch, Islamic Azad University, Yasouj, Iran

ARTICLE INFO

Article History:
Received 23 February 2022
Received in revised form
29 March 2022
Accepted 15 June 2022
Available online 16 June 2022

Keywords:
C3M Method, MoveiLens100K Dataset, Kmean Clustering Algorithm, KNN Algorithm

ABSTRACT

Today, in order to buy goods through the Internet, every company or production organization has an internal commercial software site based on which it offers its products and services to customers. To ensure that the user has the ability to provide a suitable proposal for a request or to solve a need in the midst of a huge amount of data, recommender systems are the right solution. Different methods of providing suggestions in recommender systems are divided into eight methods according to the data mining of the classification of methods. In each method, in these systems, the necessary suggestions and predictions are provided to users in a special way, and the most important method is the recommender system among other methods, is the filtering method. In this method, the number of clusters in the data set related to recommender systems is dynamically determined by c3m clustering algorithm on a data set called Movielens ml-100k, which has a data oscillator in four inputs, as well as k-means algorithm and performance optimization. It was estimated. The final clustering is done well with the help of this method, if the target user enters after the clustering operation and by matching the profile information which includes a series of items rated by similar users in the same cluster, the similar cluster search for Based on the correlation filter, which is one of the methods used by the KNN algorithm, it finds its similar cluster with each cluster head (cluster representative) and based on the items ranked in demographic information, the nearest neighbors (neighbor and similar) finds users-items and the item that has the highest rank among other users. The obtained similarity is stored in the user's top-n list and presented in the form of an offer.

1. INTRODUCTION

Ground Penetrating Radar (GPR) is a widely utilized non-invasive technology for detecting subsurface objects across various domains, including landmine detection, buried pipeline identification, and locating oil beneath snow-covered terrains. By emitting electromagnetic pulses and analyzing the reflected signals, GPR effectively identifies

* Corresponding Author: hamidreza.avini93@gmail.com
Department of Computer Engineering, Faculty of Technology and Engineering, Yasouj branch, Islamic Azad University, Yasouj, Iran

http://dx.doi.org/10.47176/TMI.2022.97

© 2022 by the authors. Licensee T.M.I, Tehran, Iran. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (http://creativecommons.org/licenses/by/4.0/).

variations in subsurface materials. Its capability to detect both metallic and non-metallic objects makes it particularly valuable in scenarios where traditional detection methods may fall short [1-2].

Recommender systems are integral to modern digital platforms, assisting users in discovering content and products tailored to their preferences. These systems operate by collecting and analyzing user data, which can be obtained through explicit feedback such as ratings and reviews or implicit feedback, including browsing history, purchase behavior, and interaction patterns. By leveraging this information, recommender systems predict user preferences and suggest relevant items, enhancing user experience and engagement [3].

Advanced recommender systems incorporate various data sources, including demographic information (e.g., age, gender, nationality) [4], social media interactions (e.g., followers, posts, likes) [5], and data from Internet of Things (IoT) devices (e.g., GPS locations, health metrics) [6]. This multifaceted approach enables the generation of personalized recommendations that account for contextual factors and user behavior.

To evaluate the effectiveness of recommendations, systems assess metrics such as accuracy, novelty, diversity, and stability. Collaborative filtering, a prevalent technique in recommender systems, identifies patterns among users with similar preferences to generate suggestions. This method is often combined with content-based filtering, knowledge-based approaches, and social filtering to enhance recommendation quality [7].

In intelligent platforms, recommender systems distinguish themselves by adapting to user preferences through continuous learning. For routine purchases, these systems analyze historical buying patterns to suggest relevant products [8]. In contrast, for specialized items, recommendations are refined through ongoing interactions, capturing evolving user interests and needs [9].

The core functions of recommender systems encompass collecting user preferences [10], constructing models to represent this information [11], managing personal data [12], and monitoring user feedback [13]. Implementing agent-based methodologies, where autonomous agents perform specific tasks concurrently, has proven effective in managing these complex processes [14]. This article provides a comprehensive overview of the research landscape, detailing algorithms, variables, and key considerations essential for advancing knowledge in the field.

2. SUGGESTED METHOD

In this section, we present and explain the proposed methodology based on the flowchart illustrated in Figure 1. Each step of the flowchart is elaborated in detail below. This research focuses on enhancing clustering-based recommender system approaches. The improvement is achieved through the integration of the C3M clustering algorithm to determine the optimal number of clusters and to perform clustering on large-scale user datasets typically used in recommender systems. The output clusters are further refined using the exclusive flat K-Means algorithm, and each stage of the methodology is systematically discussed.

As shown in Figure 1, the collaborative filtering-based recommender system (specifically, correlation-based collaborative filtering) is employed. This approach leverages the preferences and behavior of other users who have rated similar items. Throughout the process, collaborative filtering techniques are used to enhance the quality of recommendations.

Assuming the MovieLens dataset is used as the input, the first step involves data preprocessing and loading the dataset in the required format (e.g., data.mat). All data are then processed using the C3M clustering algorithm, which performs an initial user clustering. C3M is selected due to its ability to overcome certain limitations of traditional K-Means, particularly its inability to reliably handle large and high-dimensional datasets.

Following the initial C3M clustering, the exclusive K-Means clustering algorithm is applied to refine the clusters. For each target user, clusters that exhibit similar behavioral patterns are identified. This is achieved through user profiling using collaborative filtering, in combination with exclusive KNN-based clustering. This step ensures the alignment of the target user with similar users by comparing the items rated by the cluster representatives (i.e., the "head" users) against the target user's preferences.

If a match is found, the user is associated with the most similar cluster containing up to k similar users. If no suitable match exists, comparisons continue across other clusters. Ultimately, the target user is either assigned to an existing cluster or a new cluster is formed.

Once the target user is associated with a cluster, recommendations are generated by ranking the items based on user-item similarity within that cluster. The KNN algorithm is then employed to compute user ratings and sort them in descending order. The item(s) with the highest predicted rating from among similar users in the cluster are then recommended to the target user.

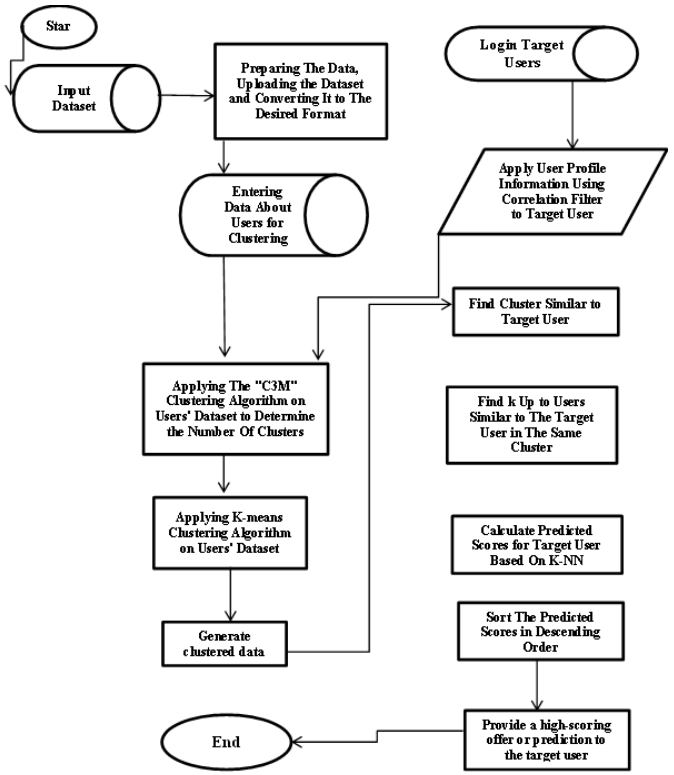

Fig.1. Suggested Diagram

2.1. Description of the proposed method

First, before explaining the proposed method, assuming the default dataset of MovieLens (the dataset for rating items related to movies by users), how to prepare and apply pre-processing on the data and prepare them to an acceptable format for the application software and MATLAB simulation, which is This research has been used.

2.2. Prepare and apply pre-processing to the data

In order to cluster users according to their demographic information (default dataset information), we need to be able to apply the data related to the relevant dataset or any other valid dataset to an acceptable format for the MATLAB simulation software used in this research. The acceptable format for the data in this software is with the extension (.mat*), to do this, first import all the default data in the Excel database software, and then import the imported file in the MATLAB emulator software in the matrix format and with the desired name. We do this in such a way that all the values of the matrix in Excel and the MATLAB simulator software are equal after extraction, if there are some nominal characteristics such as gender, occupation or any other qualitative characteristic in some datasets, they are converted into numerical data type in order to speed up the clustering. It should be noted that most of the datasets are initially in (.txt*) format, and the description of each dataset is in its readme section.

2.3. Data clustering

In the following sections, the steps to determine the number of optimal clusters and the proposed method are described.

2.3.1. c3m clustering algorithm

One of the most important phases of this research is data clustering, or in other words, to determine in a novel way which similar cluster each data belongs to, and to place similar data in the same cluster, which was discussed in the previous chapter using various clustering methods with criteria. Each clustering method was fully described.

As you can see in the diagram above, the 'c3m' algorithm is used for initial clustering of data or users. At first, using the $'c3m'$ clustering algorithm, initial clustering is done to determine the number of clusters. The use of this clustering algorithm, which among a large number of data (large (OLAP) and relatively large datasets), such as the default dataset, has the ability to determine the optimal clustering based on the density of users in a completely dynamic way among many other users. or in other words, it puts similar users in a cluster. The second reason for using this algorithm in clustering is to solve some of the disadvantages of the exclusive k-means clustering algorithm. The advantages of this algorithm over other clustering algorithms are:

It is not dependent on DATA and based on prioritization and partitioning of data, it performs similar to clustering.

- Clustering operations are distributed clusters.
- Dynamically specifies the number of k or data centre points.
- The clustering operation obtains the clusters more optimally.
- It involves less time and memory complexity.
- It is optimal and suitable for clustering a large database (database) or data warehouse (OLAP).
- Compare each data once with all the data in the form of a matrix and find the cluster belonging to it, if in other clustering algorithms such as K-MEANS based on distance, and each time to select the head of the cluster, it performs the comparison operation, which is the same. It increases time and memory and involves a lot of cost, so outlier data are very sensitive to the average and it is not a good measure of clustering.
- In this algorithm, information is retrieved every time.
- The stability of the clusters is high and it takes much less outlier data, while in k-means every time the clustering is changed according to the distance and the choice of the cluster head and it has a lot of outlier data (noise).
- In this algorithm, it performs clustering by using the correlation matrix c to match the data using query.

In this section, a pseudo-code of the $'c3m'$ clustering algorithm is presented, which is fully explained.

First, we calculate the following value for each term in each document. In another sense, we initialize the C matrix.

$$C[m,m] < -0 = "0"\ p = "p" > \tag{1}$$

For each document, it creates a matrix whose dimensions (rows and columns) are equal to $m * m$ and their values are from zero to p, and the number of corrections used in a document in the form of a matrix, which is $T = t\_1, t\_2, \dots, t\_i\}$ is (in the adjacency matrix).

$$Compute\ p(t\_i) = D(I,t)/sum\_t\{D(I,t)\} \tag{2}$$

In this part, the function calculates the probability density, which is equal to the probability of each term in an entire document on all or all documents, for example, the word to is used 10 times in a document and If this word is repeated 1000 times in all documents, the value of the probability density function shows how likely these documents are to be close to each other using this word. It obtains a matrix of all the indices of the calculated terms, or in other words, all the terms that are present in the document along with their number of occurrences in the form of a previously created m*m matrix is.

Then it calculates the probability of each document in that list (the probability of the oval rule) or in other words, how likely is it that a document in that list, which reversely matches one document with other documents each time, is the desired document be included in the list with other documents.

2.3.2. Ellipse theorem formula

$$p(h|x) = \frac{p(x|h)p(h)}{p(x)} \quad (3)$$

$$C[I,j] = c[I,j] + p(t\_i) * p(d\_j|inv(t\_i)) \quad (4)$$

To obtain each matrix, the initial value is multiplied by the Bayes probability and density probability and added to the default value, or in other words, the same value of the density function that was in the previous step in the initial value of the matrix created from a document and it is added by the probability of the default value and the document list are multiplied.

$$n\_c = \backslash sum\_\ i, c[I,j] \quad (5)$$

In this part of the code, the value obtained is the number of clusters. According to the pseudo-code above, it has a series of criteria to create matrix c, which is explained as follows:

2.3.3. Formation of matrices

As mentioned, various factors are involved in forming the correlation matrix. This matrix, which is based on correlation filtering, is used to correlate different data and create a relationship between data through spatialization (correlation similarity matrix) through the following relations, but before from each operation, the correlation coefficient relationship is as follows:

$$r = \frac{cov\ ariance\ al\ xandy}{s_x s_y} \quad (6)$$

In the formula above, s_x is equivalent to the standard deviation for s_y, x is equivalent to the standard deviation for y. To calculate the correlation coefficient, we first need to calculate and determine the covariance of two variables, which is displayed in the form of $cov(x, y)$. Covariance is similar to variance, but in covariance the deviation from the mean for x , y is simultaneously used from the following relationship:

$$cov(x,y) = \frac{\sum(x-\bar{y})(y-\bar{y})}{n-1} \quad (7)$$

Observe the similarity between covariance and variance by comparing their equations (formulas above).

It should be noted that correlation analyzes are no longer valid when causality is assumed. Correlation or regression can be calculated on a series of data, not both. Regression analysis assumes the existence of a cause and effect relationship.

The formula related to the existence of a linear relationship between two variables (regression line) is $y = a + bx$ b is the slope of the line. We divide the changes of y into x. $(\Delta y/\Delta x)$ and a is the width from the origin of the line. To determine each line, it is first necessary to calculate its slope and width from the origin:

$$b = \frac{cov(x,y)}{s^2 x} \quad (8)$$

$$a = \bar{y} - b\bar{x} \quad (9)$$

As a result, if there is a cause and effect relationship between two variables, we can predict the value of y for any given value of x, we can use the formula $y = a + bx$ or to calculate y for any desired x, we can use the regression line.

After analyzing the data, $c3m$ clustering algorithm will partition each side of the data to understand the production, which is obtained from the following relationship:

$$p = \{c_1, c_2, \dots, c_n\}, and\ \sum_{i=1}^{n_c} |c_i| = m \quad (10)$$

That the set p can be created in different ways, for example, the threshold value of a "suitable" can lead to acceptable partitioning in a single step, the average of steps, or the maximum value of a complete graph. Although the existence of partitioning in order to estimate the value It will be a threshold, but in any case, partitioning is acceptable.

In the mentioned algorithm, when a data is compared with other data each time for clustering, but using a series of balanced equations, or in other words, both sides of the equations to create a cluster (the data in a cluster that is in the software as a seed is provided in a field with specific coordinates) together with respect to other similar data:

$$d_i \neq d_i \; and \; d_i RM d_i \rightarrow d_i \in (D - D_S) \oplus (D - D_S) \tag{11}$$

Each$(D - D_S)$ represents a data sample from a cluster, which matches each data or seed similar to all the data of each cluster through $d_i RM d_i \rightarrow d_i$ There are several steps to create this method, which consists of :

- Select the first data as a cluster data.
- The data (seed) is randomly selected among a set of data (dataset).
- Data cluster generation based on random data.
- Dividing the dataset (according to the data points) into different partitions.
- For each data, according to its stability, the cluster chooses the largest data type.

To create the presented matrix according to the document and text dataset, we consider that to create the matrix provided in the document dataset assuming the number of text $\{d_1, d_2, \dots, d_m\}$ and the number of words $\{t_1, t_2, \dots, t_m\}$ In the coefficient and concept coverage matrix or matrix sc (correlation) to select the document among the documents, the corresponding domain with c_{ij}which exists $(1 \leq i, j \leq m)$which is used to select the word in each text as d_i document from d_j of the document.

In order to be able to create the matrix c with the specifications d_{ij}, $(1 \leq i \leq m, 1 \leq j \leq n)$, we must meet the following conditions, which are:

$$\sum_{j=1}^{n} d_{i,j}, 1 \leq i \leq m \tag{12}$$

Which in the above formula can display one word (similar example) for each document.

$$\sum_{i=1}^{m} d_{i,j}, 1 \leq j \leq n \tag{13}$$

In the formula above, each word is to be entered into a document for its clustering. In general, we have:

$$s_{i,j} = d_{j,k}/(\sum_{h=1}^{n} d_{i,h}), \overline{s_{j,k}} = d_{j,k}/(\sum_{h=1}^{m} d_{hk}) for \; 1 \leq i \leq m, 1 \leq k \leq n \tag{14}$$

Which creates the following matrix:
If the vector X is our random variables, then we have:

$$x = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \tag{15}$$

The covariance matrix is a matrix whose terms are obtained in the following way.

$$E_{i,j} = cov(x_i, x_j) = E(x_i - \mu_i)(x_j - \mu_j) \tag{16}$$

which gives the following matrix:

$$\sum \begin{bmatrix} E[x_1 - \mu_1)(x_1 - \mu_n)] \dots E[(x_n - \mu_n)(x_n - \mu_n)] \\ E[x_n - \mu_n)(x_1 - \mu_1)] \dots E[(x_n - \mu_n)(x_n - \mu_n)] \end{bmatrix}$$

where E is the mathematical expectation.

If the covariance of two random variables is zero, then the two variables are called uncorrelated. If two random variables X, Y are independent, then their covariance is zero. This issue can be concluded as follows:

Because: $E(X.Y) = E(X).E(Y)$, then we have $COV(X,Y) = 0$, while the opposite of this issue is not true, that is, the covariance of two random variables may be zero, but those two variables are not independent by chance.

In the matrix above, as you can see, it deals with the correlation between the data, or in other words, for the relationship between two interdependent variables, such as the height of a person compared to its weight, it determines the level of health or its appropriate weight. In this matrix, for the relationship between Different data for optimal clustering at the right time, we use the c matrix that various factors and formulas make this matrix, one of these formulas is the covariance matrix that performs the data clustering operation and a proximity matrix that makes it possible to distinguish between different data. Determine the number of clusters in the datasets, the criteria for forming this matrix are described in the rest of this research.

2.3.4. K-means clustering algorithm

In the previous section, we fully explained the initial clustering of data on the default dataset, and on the other hand, the number of clusters was determined dynamically, then the exclusive K-means clustering algorithm performs the final clustering on the desired data or users. Or in other words, by obtaining the number of clusters in the $c3m$ clustering algorithm, in this distance-based algorithm that uses the average measure to determine the representative of each cluster randomly, it performs the final clustering operation on the data or users. The K-means clustering algorithm has a series of disadvantages that were stated in the previous chapter, most of which were solved with the c3m algorithm, because the final clustering algorithm is formed in this algorithm and assuming a set of target users who use the information Profiling of each user is done after the final clustering using collaborative filtering algorithm (correlation) based on exclusive K-means clustering algorithm, or in other words, after the final clustering operation if there are target users according to the demographic information and scoring method of each user. matched with the representative of each cluster to find the number of k users similar to the target user, which will be discussed further in the next sections.

2.4. Simulation steps and proposed method

In this part, we will present the simulation step by step in the form of images and diagrams. The first step in the proposed method is the prefilling and data preparation step.

2.4.1. Simulation of the preprocessing step of the data source

The data of this research has been presented with the extension.mat, which has been converted to.mat in various formats such as .txt, .xlsx for the many necessary uses of the extensions. For this purpose, it is necessary to have a dataset related to the users in the format. mat. First, the default dataset (ml-k100 dataset) in .txt format has been extracted using WordPad software in Excel software with .xlsx format. The use of clustering algorithms, which are fully described step by step in the following sections, is used. In the proposed method, which is an optimal and useful method, the dataset related to the default recommender systems of users-items is used for clustering with the name ml- k100 dataset is as follows.

The default dataset of users-items, which is a dataset in the form of one hundred thousand data in four entries. In the default dataset, a series of training and test data is used in several related files, the overall file of which is in four entries, which are:

- User ID: row number of existing users (educational) in the dataset.
- Item ID: item number (film and its specifications) available in the dataset.
- Rating: Educational users' score for each item in the dataset.
- Time Stamp: time stamp related to the items and received by each educational user in the dataset.

The total data is a total of 100,000 points (range 1-5 points) by 943 users on 1682 items, each user has rated 20 movies (items). are arranged There are time stamps from 1970/1/1 UTC of Unix seconds.

Finally, a dataset with high specifications is prepared to be applied to clustering algorithms through simulation software, which is prepared as a $100000 * 4$ matrix in the Workspace section of this software and is applied to clustering algorithms that are fully described in the following sections.

2.4.2. Initial clustering of users

After the data has been pre-processed, initial clustering should be done. As explained in the previous chapter, the k-means clustering algorithm performs the clustering operation based on the average, and it had some disadvantages, but in the proposed method, now with The use of c3m clustering algorithm which overcomes the disadvantages of k-means clustering algorithm, which among the most important tasks of this algorithm is the dynamic calculation of the number of clusters in the dataset related to the recommender systems, which in case of adding a new user using demographic information Based on correlation filtering, it clusters its similar cluster or, in other words, similar users with the new user, which explains the clustering operation step by step in the following sections.

The c3m algorithm is a type of single-stage clustering algorithm for data clustering. The basis of this algorithm is to select a set of data (documents, movies, music, images, sites, etc.), considering each data as a point. that dependent or similar clusters are placed in one cluster, and non-dependent clusters are placed in other clusters, that the clustering operation procedure in this algorithm has several steps as follows:

- Creating a general and basic correlation matrix to identify similar data
- Calculate the power of each point in the matrix as a data in each cluster
- Separation of related and similar data in the desired cluster

Now, in the implementation phase of the relevant data according to the desired algorithm in the simulator software, we have the primary data in the form of the diagram below, where in the above matrix similar and dissimilar users and the location of each user on the diagram were determined according to the numerical data that each the user is placed as a point in the following diagram.

As you can see in figure (2), the desired data of the location of each of their coordinates on the graph was determined that the desired data is related to the movie recommender systems, and the relevant data set contains one hundred thousand points (range of points from 1 to 5) from 943 users are on 1682 movies (items). Each user can assign different points to 20 desired items and it is in the form of a 100000*4 matrix, or in other words, one hundred thousand data in four inputs. In this dataset, as mentioned, in the way of scoring in the algorithm, the proposed method first displays the users in a vector. Clustering in the form of users with similar interests, or in other words, users who have common interests and tastes for each item (movie) in the diagram below shows the output related to the clustering of the c3m algorithm on the above data. It dynamically performs the clustering operation using Gaussian functions and detects the number of clusters in the user-item dataset, which is in the form of the diagram below.

In Figure (6), as you can see, the distance and iteration criteria of the output related to the users-items dataset and the movie recommender system, which was prepared in the form of a matrix of numbers in the pre-processing stage, and according to diagram 4-4, it has a series of points (seed) appears in different parts of the diagram and is clustered, and each of the points has a coordinate corresponding to the points to find the head of each cluster using the desired loop in the program until the head of the cluster does not change and for each point its distance to The center of the cluster shows the repetition of the best time to repeat the data in the desired cluster. Then it should be mentioned that the best repetition of the data in the cluster is the ratio of the distance to its coordinates in a time equivalent to (Time = 263.8836) and (Best Fitness = 23293196317.7663). Then, in the next part, we will talk about the target user and how to introduce it to the new cluster and the nearest neighbors, which will be fully examined and explained in the next part. The evaluation criteria in relation to the coordinates in the dataset is as follows.

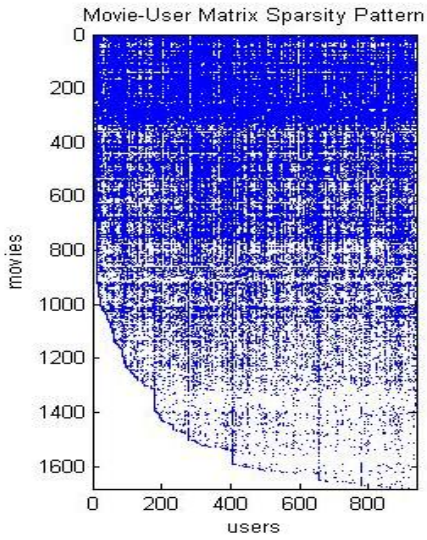

Fig. 2. Specifying the coordinates of each point of the default users-item dataset before clustering in the c3m algorithm.

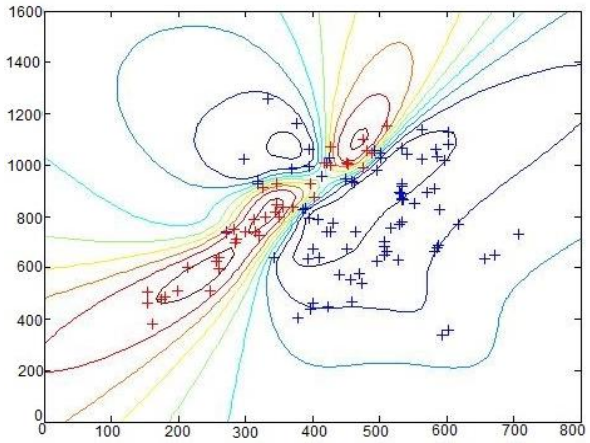

Fig. 3. Initial clustering diagram of default dataset by c3m clustering algorithm.

As you can see in Figure (3), the number of clusters related to the default dataset was determined dynamically. In the above algorithm, the number of clusters was clustered into 19 clusters using the desired unsupervised algorithm based on Gaussian and density functions. The clustering operation facilitates the clustering in the k-means algorithm and improves and reduces the time and memory complexity, which in the next section we describe the implementation of the k-means algorithm on this default dataset with the number of clusters.

2.5. Applying the k-means clustering algorithm to the data

The c3m clustering algorithm has performed the primary clustering operation on the data, as mentioned, one of the most important tasks of this clustering algorithm is to dynamically calculate the number of clusters, since the k-means clustering algorithm is based on the average and applying the user dataset to On this algorithm, it is able to perform clustering based on the distance of users to each other. If after obtaining the number of clusters, using the c3m clustering algorithm dynamically and fixing the disadvantages and optimizing it as mentioned in the previous sections It performs the clustering offline, and in case of adding a new user and finding a cluster similar to the new user, it performs the correlation based on filtering, and the result is shown in the following figures in the output of this algorithm in relation to the user dataset. It should be noted. that the clustering operation of this algorithm is fully

explained. For the clustering of educational users, first before clustering, the location of each data, where each data is considered as a user, is in the following diagram.

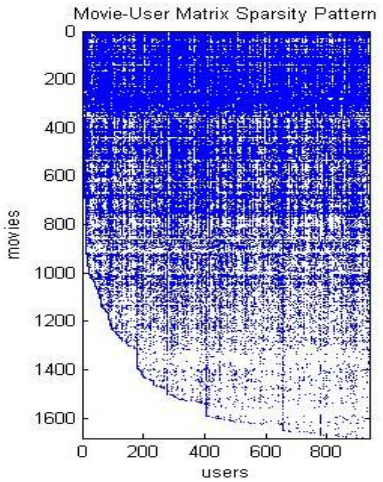

Fig. 4. Specifying the coordinates of each default data point which is a matrix of 4x100000 in the graph before clustering in the k-means algorithm.

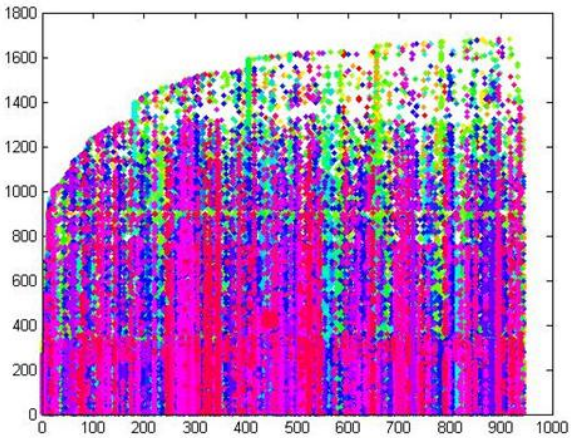

Fig. 5. Data clustering after initial clustering to determine the number of clusters by k-means algorithm.

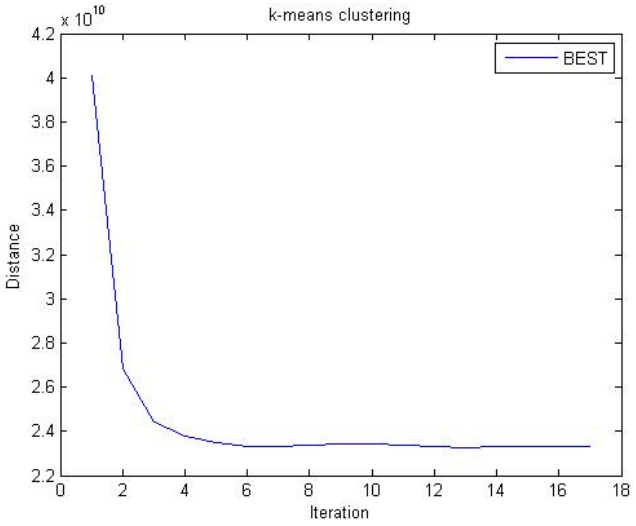

Fig. 6. Diagram of repetition of the best data in each cluster and relative to its coordinate distance.

2.6. Applying the K nearest neighbor algorithm to the data (KNN)

The types of evaluation criteria in this algorithm in the material software are divided into two parts, which are: 1- The amount of regression error 2- The amount of error based on cross-validation, which in the first part calculates the amount of data error in terms of regression, which the number of errors If it is less, we have better data, or in other words, the fewer the number of errors in the cluster, that cluster is considered as the best cluster. In the k-nearest neighbor algorithm, this method is used less, but in the second method, during data clustering, this The algorithm should first check what behavior it can have with the data.

2.7. K-Fold

In this type of validation, the data is divided into K subsets. From these K subsets, each time one is used for validation and another K-1 is used for training. This procedure is repeated K times and all data are used exactly once for training and once for validation. Finally, the average result of these K times of validation is chosen as a final estimate, and the implementation of the steps is as follows. Comparing the clustering error by the number of clusters: with the above explanation, taking into account the amount of data error in the default dataset in each cluster. The output diagram is as follows. In the KNN algorithm, to display the graph of the errors related to each cluster in relation to the training data after loading the desired data dataset and display the values as an n*m matrix in the command window part of the simulation software.

```
>> load fisheriris
>> X=meas

X =

    5.1000    3.5000    1.4000    0.2000
    4.9000    3.0000    1.4000    0.2000
    4.7000    3.2000    1.3000    0.2000
    4.6000    3.1000    1.5000    0.2000
    5.0000    3.6000    1.4000    0.2000
    5.4000    3.9000    1.7000    0.4000
    4.6000    3.4000    1.4000    0.3000
```

Fig. 7. Loading data in the default dataset for the number of errors in different ways in the KNN algorithm.

As you can see in Figure (7), one hundred thousand data are displayed in four inputs. Also, after performing the operation of specifying clusters by c3m clustering algorithm and data clustering by K-MEANS algorithm, naturally, the number of errors in each dataset related to the recommender systems is determined dynamically by using

Gaussian and density-based functions. It will be less on the training data available in the dataset, and if test data is added to the data using profiling information in the form of a matrix in the form of a series of numbers and specifying it on a graph based on correlation filtering which is done through one of its own methods called The KNN algorithm is introduced to the new data with the closest and most similar data, which includes a predefined list that includes a series of items that have the highest scores among other items and is placed in the TOP-N list, and presents a proposal to the new data. and it is clustered with other similar data, and this clustering is in the form of a new offer to the new user (target user) and is based on the mutual filtering of all future offers that are introduced to the data in that cluster. The newly arrived user, who is also present in the training data cluster, is introduced in the MOVIE LENSES dataset related to users and movies, based on a series of items related to each movie in the form of a matrix. Among the users, it is ranked as the best item and it is placed in the TOP-N list, which is compared with the head of each cluster using profiling information of each user using correlation filtering, and in case of similarity, among The members of the cluster find the nearest neighbors, such as the TOP-TOP or KNN algorithm, the corresponding cluster is determined, and it is clustered with other similar users based on their interests and tastes related to movie selection. At first, it displays the user-item dataset diagram, and then In order to specify the target user, it first displays the current precision chart.

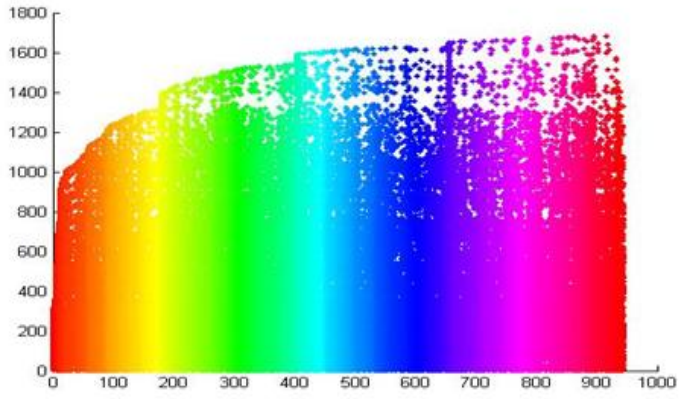

Fig. 8. Determining the target user cluster among other clustered training users based on correlation filtering.

In the previously mentioned user-item dataset, each user gives special points to a series of items (movie), which are the target user among other educational users based on collaborative filtering, which means that the system cooperates and contributes to all users. (Scoring each item) and the item that gets the most points among most and sometimes all users as the best item in the TOP-N list in the suggester system for new suggestions to target users according to their interests and the users' tastes are sent in each similar cluster, which is presented to the target user in the following output in the corresponding dataset of users who have the same interest and who have gained the most points, as shown in the diagram below.

In Figure (9), looking at the slope of the fit lines, it can be said that Kevin and Jay as users do not share the same goals because their ratings are negative. One of the common measures of similarity in collaborative filtering is the Pearson correlation coefficient. This variable ranges from 1 to -1, where 1 is a positive correlation, 0 is no correlation, and -1 is a negative correlation. We calculate the correlation of users using rows without any missing values. The similarity scores between Kevin and Jay are equal to -0.83 and Kevin and Spencer are equal to 0.94. Since Kevin and Jay have different tastes, their similarity is negative. Kevin and Spencer, on the other hand, share very different tastes. Similar users are called neighbors, and we can predict the ratings of arbitrary items by combining our existing ratings for other items. But we need to find these neighbors first. Let's find neighbors for Kevin. Kevin has three neighbors who get along well with him. We can use our rankings and correlation scores to predict a coin's ranking. The weighted average method is a basic approach for forecasting. Because the rating scale can vary among individuals, we need to use centered ratings instead of raw ratings. Kevin has not yet rated 'Boyhood'.

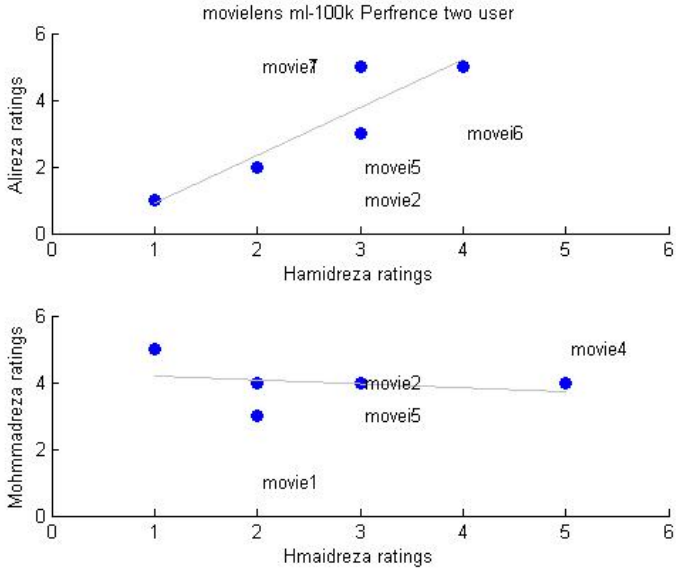

Fig. 9. Rating diagram of users who have the most votes and similarities in the users-item dataset.

Table 1. Ranking of users in the form of a matrix according to similarity with each other.

	Actual	**Predicted**
Movie1	2	1.3103
Movie2	3	1.1748
Movie3	NaN	3.9652
Movie4	5	4.2751
Movie5	3	3.3103
Movie6	4	3.6878
Movie7	2	2.3058
Predicted rating for "Movie3": 4		

According to the above explanations about the MOVIE LENS dataset, the precision diagram for the proposed method is as follows.

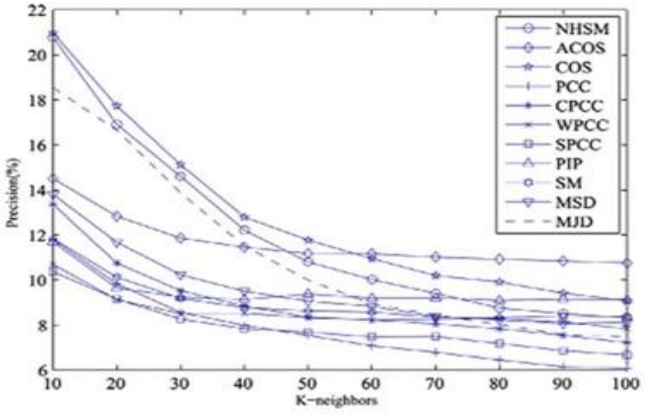

Fig. 10. The diagram of the precision diagram of the users-related item dataset based on collaborative filtering.

In the diagram below, there are measurement criteria as follows:

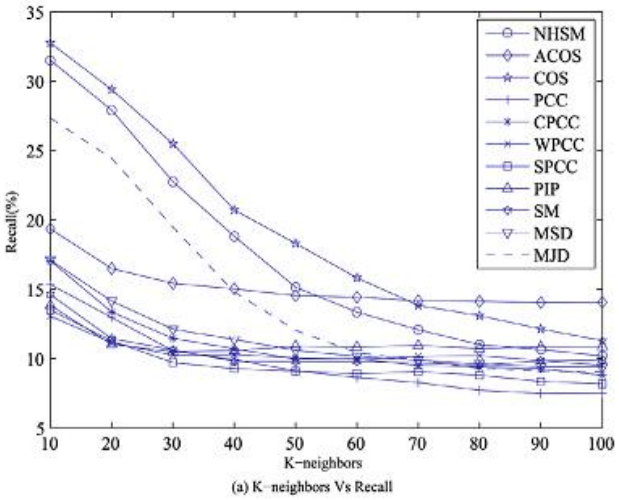

Fig. 11. Nearest neighbor diagram to k along with recall diagram.

The MAE diagram related to the simulated dataset is as follows.

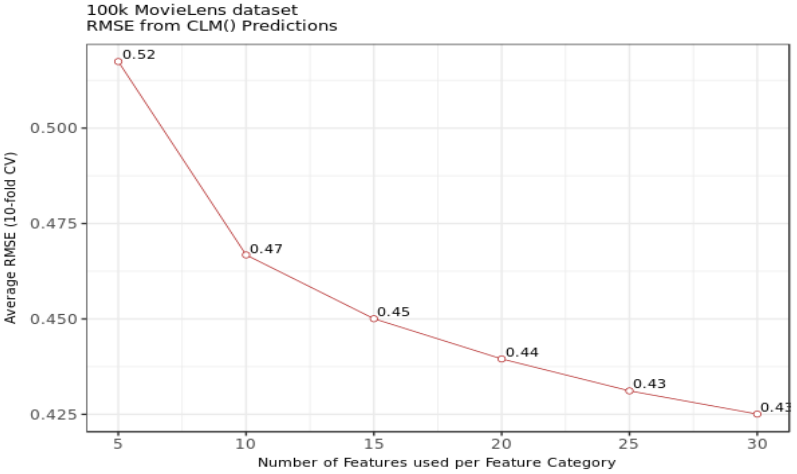

Fig. 12. Average RMSE diagram of 10 times validation of clm with 5, 10, 15, 20, 25, and 30 nearest neighbors.

Figure (12) gives the average RMSE of CV 10 times after selecting 5, 10, 15, 20, 25, and 30 nearest neighbors from the user and the proximity and item similarity matrix. Once again, similarity matrices represent memory-based information, while proximity matrices carry content-based information. Four different sources of features are called "feature categories" in the diagram, and it should be noted that the average error rate is 0.5567.

Results of likelihood ratio tests are presented, along with Akaike information criterion (AIC) values for all models; A comparison between AIC values is presented. I compare models that include top neighbors from memory and content-based matrices, while models that only include information from memory-based matrices on the other hand. As you can see, AIC is consistently higher for models which include content and memory-based information (with significant LRTs showing that the contribution of content-based information cannot be rejected) is lower.

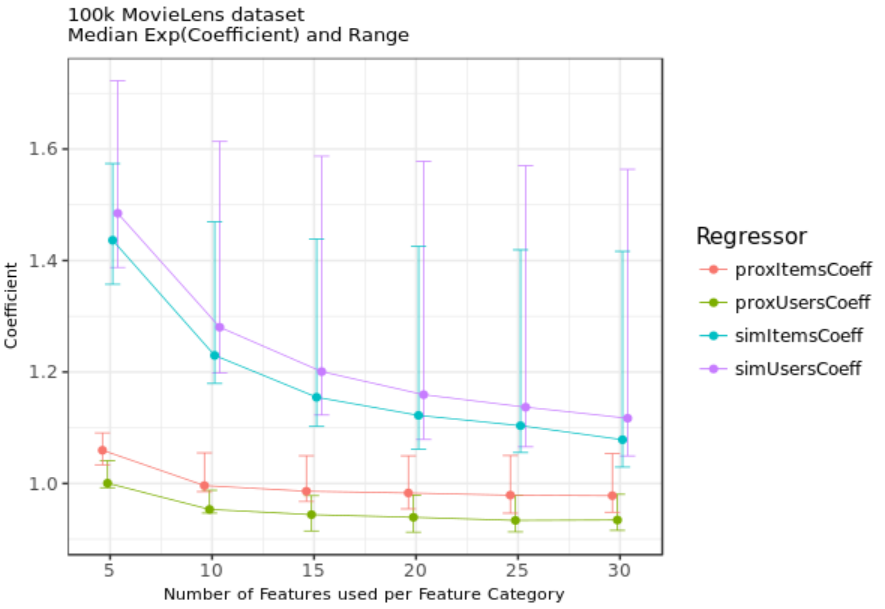

Fig. 13. Average diagram (β) of CLM estimated instructions of the logit model without cross-validation and including 5, 10, 15, 20, 25, and 30 nearest neighbors from the matrix based on memory and content.

The following diagram is related to the MAE diagram of the relevant dataset:

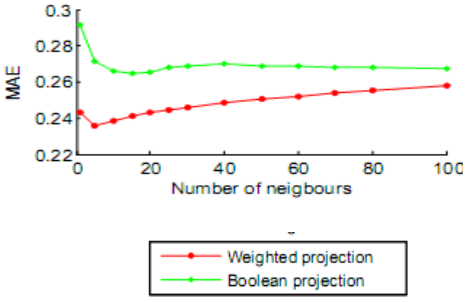

Fig. 14. MAE diagram related to MOVIELENS.

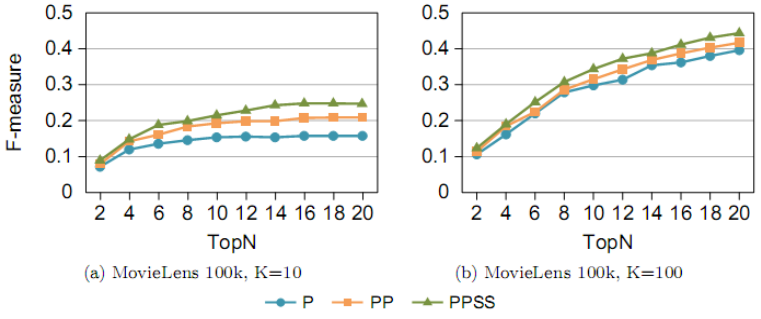

Fig. 15. F-measure diagram related to Movielens dataset.

2.8. Comparing the results of the proposed method with other methods

In this comparison, firstly, the investigation of the evolution of accuracy and coverage of a different pattern when the percentage of data as a training set has been changed. The results are shown for the MovieLens data, while in this comparison, another comparison is shown in the Netfix stream. As the percentage of data in the training set (dataset) increases, the users' rating matrix also increases, and naturally, the error is higher and more information is required. The prediction is calculated. Similarly, with the increase of information density, a slight decrease is observed among the algorithms, while the difference between each of them is in the density of information density. A comparison among different algorithms such as (UB, RSVD2[†], SVD[‡]++, RSVD, SO[§], TB [**]while at 10%, SVD ++ provides the best results along with RSVD2, TB, NSDV2, RSVD.

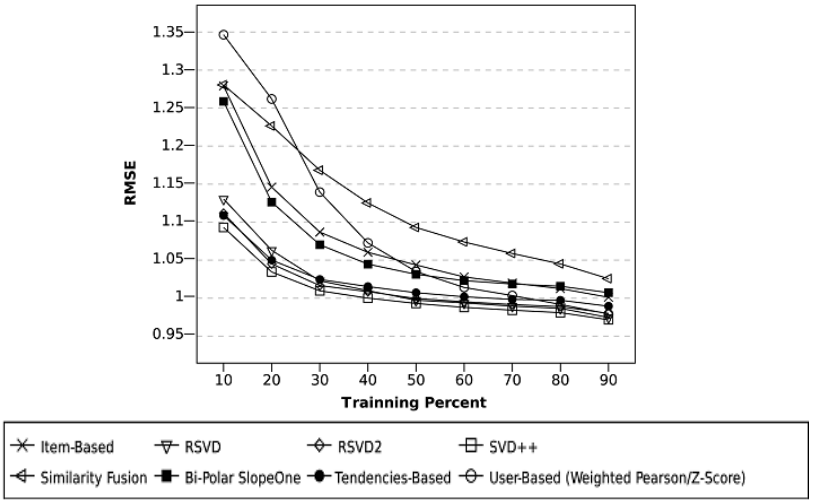

Fig. 16. RMSE evolution diagram according to matrix density, for MovieLens data set.

As can be seen in Figure (16), the RMSE diagram of the algorithms in question was evaluated based on different similarity criteria in the Movielens dataset, which includes the error range of $0.95<x<1.05$. Although we do not deny this limitation, our results show that, on the contrary, this is the biggest limitation of an algorithm affected by lack of information.

In other words, the prediction accuracy is limited by the amount of information that can be obtained from the ranking matrix. And this is mostly limited by the information actually in the matrix, that is, its density. In the RMSE diagram below, our values before the corresponding dataset are as follows.

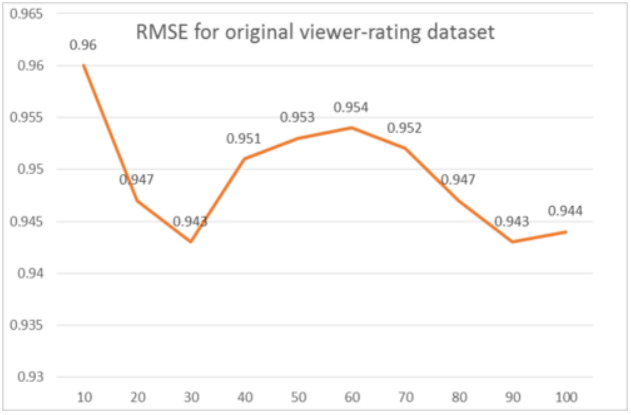

Fig. 17. RMSE diagram of former methods for other datasets.

[†] - Randomized Singular Value Decomposition.
[‡] - Singular Value Decomposition.
[§] - Slope one.
[**] - Tight-Binding.

In this section, as reviewed, the algorithms that were compared with the proposed method in the form of a general diagram relative to the item or method, the MAE [††]error rate is as follows.

Table 2. Computational ability of different studied algorithms

Algorithm	Training	Prediction
User-based	-	**O(mn)**
Slope one	$O(mn^2)$	**O(n)**
Tendencies-Based	$O(mn)$	**O(1)**
RSVD	$O(mnk)$	**O(1)**
RSVD2	$O(mnk)$	**O(1)**
SVD++	$O(mn^2k)$	**O(1)**
NSDV2	$O(mnk)$	**O(1)**

Table 3. Comparison of the proposed method with previous methods.

SVD ++	RSVD	RSVD2	SO	UB	suggested method
0.73	0.75	0.68	0.63	0.78	**0.43**

As you can see from Table 3, the error value of the proposed method is less than other methods with a value of 0.43, So the proposed method can be trusted and used in the proposed systems.

3. CONCLUSION

In the contemporary digital age, the rapid growth of the global population and the increasing diversity of individual needs have led to a growing demand for personalized services. Simultaneously, the continuous expansion of the web and the widespread integration of intelligent systems into websites have enabled platforms to analyze vast volumes of data and generate personalized predictions and recommendations for each user. Given the massive scale of data such as those found in OLAP data warehouses, large-scale datasets, and extensive databases recommender systems have emerged as an effective solution for delivering targeted suggestions and insights tailored to user preferences.

Various clustering methods have been employed in recommender systems to manage and organize large datasets. However, each method presents its own set of limitations. To address these challenges, the proposed approach adopts the Cover-Coefficient-Based Clustering Method (C3M) as an initial step. This technique dynamically determines the optimal number of clusters and is used to overcome the shortcomings of traditional clustering methods such as the exclusive K-Means algorithm.

While K-Means a popular partitioning method in clustering is commonly used due to its simplicity and efficiency, it often fails to perform adequately on large-scale or complex datasets. The proposed method enhances K-Means by integrating it with C3M, which serves as a density-based clustering technique that not only optimizes the performance of K-Means but also improves the accuracy and reliability of clustering in large databases and data warehouses. Looking ahead, the integration of innovative designs and advanced strategies holds promise for further improving the performance and adaptability of the proposed recommender system framework.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

REFERENCES

[1] Chowdhury, S. B. R., Ghosh, S., Li, Y., Oliva, J. B., Srivastava, S., & Chaturvedi, S. (2021). Adversarial

[††] - Mean Absolut Error.

scrubbing of demographic information for text classification. Conference on Empirical Methods in Natural Language Processing (EMNLP).

[2] Oncioiu, I., Căpușneanu, S., Topor, D., Tamaș, A., Solomon, A., & Dănescu, T. (2021). Fundamental power of social media interactions for building a brand and customer relations. Journal of Theoretical and Applied Electronic Commerce Research, 16(5), 1900–1914. https://doi.org/10.3390/jtaer16050096

[3] Phadnis, A. (2018). The internet of things. International Conference on Communication Systems and Networks (COMSNETS).

[4] Alqudsi, Y., Alsharafi, A. S., & Mohamed, A. (2021). A review of airborne landmine detection technologies: Unmanned aerial vehicle-based approach. 2021 International Congress of Advanced Technology and Engineering (ICOTEN). https://doi.org/10.1109/ICOTEN52080.2021.9493528

[5] Zhang, X., Ruan, C., Wang, W., & Cao, Y. (2021). Submersible high sensitivity microwave sensor for edible oil detection and quality analysis. IEEE Sensors Journal, 21, 13230–13238. https://doi.org/10.1109/JSEN.2021.3067933

[6] Shrestha, A., Vassileva, J., & Deters, R. (2020). A blockchain platform for user data sharing ensuring user control and incentives. Frontiers in Blockchain, 3, Article 497985. https://doi.org/10.3389/fbloc.2020.497985

[7] He, R., & McAuley, J. (2016). Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. Proceedings of the 25th International Conference on World Wide Web, 507–517. https://doi.org/10.1145/2872427.2883037

[8] Lingras, P., Haider, F., & Triff, M. (2017). Fuzzy temporal meta-clustering of financial trading volatility patterns. Proceedings of the 2nd International Conference on Fuzzy Systems and Data Mining, 219–238. https://doi.org/10.3934/bdia.2017018

[9] Perumal, S. P., Sannasi, G., & Arputharaj, K. (2019). An intelligent fuzzy rule-based e-learning recommendation system for dynamic user interests. Journal of Supercomputing, 75, 5145–5160. https://doi.org/10.1007/s11227-019-02791-z

[10] Sun, K., Qian, T., Chen, T., Liang, Y., Nguyen, Q., & Yin, H. (2020). Where to go next: Modeling long- and short-term user preferences for point-of-interest recommendation. Proceedings of the AAAI Conference on Artificial Intelligence, 34(1), 214–221. https://doi.org/10.1609/aaai.v34i01.5353

[11] Mukhamediyeva, D. T., & Niyozmatova, N. (2019). Problems of constructing models of intellectual analysis of states of weakly formalizable processes. Journal of Physics: Conference Series, 1210, 012102. https://doi.org/10.1088/1742-6596/1210/1/012102

[12] Birch, K., Cochrane, D., & Ward, C. (2021). Data as asset? The measurement, governance, and valuation of digital personal data by Big Tech. Big Data & Society. https://doi.org/10.1177/20539517211017308

[13] Malik, T., Ambrose, A. J., & Sinha, C. (2021). Evaluating user feedback for an artificial intelligence-enabled, cognitive behavioral therapy-based mental health app (Wysa): Qualitative thematic analysis. JMIR Human Factors. https://doi.org/10.2196/preprints.35668

[14] Cavalcante, R. A., & Roorda, M. (2013). Freight market interactions simulation (FREMIS): An agent-based modeling framework. Procedia Computer Science, 19, 680–687. https://doi.org/10.1016/j.procs.2013.06.116