International Journal of
Mathematical Modelling & Computations
Vol. 10, No. 04, Fall 2022, 225- 247
DOI:10.30495/ijm2c.2023.1962562.1256

A Novel Method for Improving Cold Start Challenge in Recommender Systems through Users Demographics Information

T. Abedini[a], A. Hedayati[b,*] and A. Harounabadi[b]

[a]*School of Management and Economics, Islamic Azad University Science and Research Branch, Tehran, Iran,*
[b]*Department of Computer Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran.*

Abstract. The user cold start challenge, occurs when a user joins the system which used recommender systems, for the first time. Since the recommender system has no knowledge of the user preferences at first, it will be difficult to make appropriate recommendations. In this paper, users demographics information are used for clustering to find the users with similar preferences in order to improve the cold start challenge by employing the kmeans, k-medoids, and k-prototypes algorithms. The target users neighbors are determined by using a hybrid similarity measure including a combination of users demographics information similarity and users rating similarity. The asymmetric Pearson correlation coefficient utilized to calculate the user rating similarity, whereas GMR (i.e., global most rated) and GUC (i.e., global user local clustering) strategies are adopted to make recommendations. The proposed method was implemented on MovieLens dataset. The results of this research shows that the MAE of the proposed method has improved the accuracy of the proposals up to about 26% compared to the GMR method and up to about 34% compared to the GUC method. Also, the results show about 60% improvement in terms of rating coverage compared to the GMR and GUC methods.

Received: 05 July 2022, Revised: 20 November 2022, Accepted: 20 December 2022.

Keywords: Recommender systems; User cold start; Clustering; Hybrid similarity measure; Asymmetric pearson.

AMS Subject Classification: 34D23, 92D30, 92B05, 93A030.

Index to information contained in this paper

*Corresponding author. Email: hedayati@iauctb.ac.ir

1. Introduction

With the development of the Internet, users are now faced with big data, forcing them to spend a lot of time to find the specific piece of information they need. Recommender systems are software tools and techniques for filtering and analyzing these data. The concept of a recommender system (RS) is utilized for finding ways to recommend items based on the tendencies of the users. Predicting and recommending the items[1] which can most probably draw a particular users attention, these systems help users find, access, and select their items of interest. These recommendations are based on the likes and dislikes of users. When users look for a particular item, they expect to find the items that existed without their knowledge or the items that they did not know how to search for. The recommender systems is looking for ways to offer items based on users interests ([31], [30], [35], [15], [29], [17]). In addition, every user can view only a limited number of these items. Hence, recommender systems identify the user needs to find the relevant items and help deal with the information overload problem. In fact, these systems are known as an indispensable component of e-commerce that can facilitate the process of business smartification in recent decades [18].The most popular recommender systems are the ones used by Amazon, Netflix, Google News, and Facebook [28]. To make recommendations, these systems need to obtain data explicitly (user ratings for items) or implicitly (tracking user behavior, purchase history, browser data, downloaded programs, reviewed books, and purchased songs) from users [28].The recommender systems analyze the resultant data to make the recommendations that can match user preferences and to predict user ratings for items in the future. Recommending the user-required items in compliance with user preferences can potentially convince them to revisit the website and make another purchase. Hence, these systems play a central role in the world of online shopping, for they help increase sales, improve diversity of sales, boost income, and lead to customer satisfaction ([29], [36], [21]). For data analysis, recommender systems benefit from four common methods including content-based filtering, collaborative filtering, knowledge-based filtering, and hybrid filtering ([31], [30], [35]). The content-based filtering recommender systems analyze the features of items or characteristics of users to make predictions and recommendations. Therefore, the recommended items are usually similar to the items in which users were interested in the past. In the collaborative filtering systems, recommendations are based on a users behavior or his/her behavioral similarity to the other users [25]. CF techniques are based on the quality of the item rated by the users neighbors. Since the user already has a level of interest in different items, CF techniques can recommend items with different contents [31]. In fact, collaborative filtering is based on the idea that if an active user shared interests with particular users in the past, the other recommendations made by similar users should be relevant and also liked by the active user [29]. The knowledge-based filtering systems use the knowledge or information regarding users, items, and their mutual relationships. Describing how to meet user needs through an item, these systems require specific knowledge of users and items [28]. The hybrid filtering systems consist of a combination of the previous systems.

The collaborative filtering technique is the most popular and widely-used recommender method. This technique finds the neighbors of a user (i.e., the neighbors are the users that share similar rating histories with the current user) and then employs their ratings to make recommendations ([31], [12]). Generally, the collaborative filtering technique is divided into three major methods called the memory-based

[1]is a general term used to show what the system recommends to users.

filtering technique, the model-based filtering technique, and the hybrid filtering technique ([12], [36]). The memory-based collaborative filtering algorithm determines the similar relationships between users and items in accordance with the user-item rating matrix and then recommends the items that are rated highly by similar users [12]. Memory-based methods that directly use stored scores for estimation [30]. The memory-based filtering technique is often divided into the user-based filtering and item-based filtering techniques [36]. In the user-based collaborative filtering technique, users with similar historical ratings must have similar interests; therefore, it is possible to predict an active users rating on particular items with respect to the ratings of similar users on similar items [7]. The item-based collaborative filtering technique calculates the recommendations based on the similarity between items but not between users. The user-item matrix is utilized in this technique to determine how the retrieved items are similar to an item of interest, measure their corresponding similarities, and make some recommendations to the user [21]. The model-based recommender systems employ the user ratings and preferences to learn a model about the user in order to make new recommendations to the user ([30], [16]). The process of learning this model is performed through data mining and machine learning techniques [31]. The key advantage of the collaborative filtering technique is that it analyzes only the mutual relationships of individuals without considering the features of items; thus, there is no need for any knowledge about the real context of items to make recommendations. In other words, such systems can make predictions without perceiving a movie, a friend, or a piece of music. Hence, it is possible to use this technique extensively without considering the content of data ([16], [20]). However, since the collaborative filtering systems are based on the comments, ratings, and behavioral similarity of each user in relation to other users, they may face some challenges if there are only a few comments by other users or if the rating matrix is empty. Caused by the sparseness or insufficiency of data, the cold start challenge is among the serious problems with such systems ([31], [30], [35], [4], [10]). This challenge occurs when a new item is added to the reserves of a recommender system or when a new user starts interacting with the system. In this case, the collaborative filtering recommenders face a serious challenge in making recommendations ([17], [16], [34], [22]). Therefore, neither can the preferences of new users be predicted, nor can the new items be rated or purchased by users, something which would lead to inappropriate and inaccurate recommendations.In this study, the proposed method analyzes the new user cold start challenge. For this purpose, the novel hybrid method is based on the collaborative filtering technique and user demographics information. The proposed method consists of two main phases. First, users information is clustered with respect to their demographics information to find the users who are similar to the target user and reduce the user information problem space complexity based on user demographic data. After that, a novel hybrid similarity measure is employed to find strong neighbors related to the target user. This similarity measure consists of rating similarity of neighbors based on the asymmetric Pearsons correlation and the demographic similarity of users. Finally, the target users rating is predicted by combining the GMR (i.e., global most rated) strategy (the highest rating given by neighboring users) and the GUC (i.e., global user local clustering) strategy (the similarity of resultant movies to the movies of the dataset) [35] . The proposed method is characterized by the following innovations:

- Proposing a novel hybrid method based on the combination of collaborative filtering and user demographics information to improve the cold start challenge
- Employing different clustering algorithms to reduce the problem space complexity in finding similar users

- Finding efficient similarity between the target user and the other users by creating a hybrid similarity measure.
- Making recommendations to the target user with the purpose of combining GMR and GUC strategies.

In this paper, Section 2 review the related works, and Section 3 presents the proposed method. In Section 4, the results of the proposed method are evaluated and compared with those of the related works. Finally, Section 5 draws a conclusion and presents the future suggestions.

2. Related works

The current recommender systems are faced with the increased amount of data, heterogeneity of data, and insufficient adaptation of data to the needs and interests of users. Thus, studies have focused particularly on the ability to aim the necessary user information. In fact, these systems aim to reduce the information overload through the process of collecting, filtering, and recommending items and information needed by users. Many of the existing algorithms still suffer from the sparseness problem, which is caused by the insufficient quantity of interactive data and feedback. Previous studies proposed different methods for solving this challenge. Some studies focused mainly on the discovery of additional data to compensate for the scarcity of data. However, some other studies focused only on similarity measures for accurately determine the similarity between users and items. In addition, a few other studies proposed a strategy for improving the cold start by using hybrid methods and data mining techniques. In [32], users demographics information were employed instead of the rating history to solve the cold start challenge and compensate for the scarcity of data. For this purpose, the frequency of each demographic characteristic was calculated to make recommendations accordingly. According to the results, demographic characteristics had equal effects on recommendations. The Disadvantages of this research is not considering the content of the items and the effect of users' ranking and also in this method, the entire data set is examined each time, so as the data set grows, the computational speed decreases. In [19], demographic data were used to identify users with similar behavior through a weight similar measure and the classification technique for mitigating the cold start in recommender systems. The disadvantages of this method are the complexity of the calculations and the lack of use of an exact similarity measurement. In [13], the ratings of reliable users were utilized to solve the sparseness of data and deal with the cold start challenge. For this purpose, the target users reliable neighbors were identified first. After that, the ratings of orthogonal neighbors for every item that was rated by at least one orthogonal neighbor were compared with the target users rating in a single value. Finally, a new user index was created with respect to the rating information. The advantage of research results indicated that finding the target users reliable neighbors improved the cold start challenge; The disadvantages of this research is that the contents of items were not considered in that study to make recommendations and this research does not consider users demographic information. In [24], the additional information of items beyond the user-item rating matrix was used to improve the sparseness challenge. The study collected the additional information about movies from `https://www.imdb.com`. For this purpose, the user index was created with respect to the user interests in items after the additional information was collected and integrated (e.g., users, items, user-item matrix). Finally, a classifier was utilized to predict user ratings for items. The advantage of research results demonstrated that sparseness was reduced

effectively; The disadvantage of this research is that no acceptable results could be obtained in the absence of additional information or the presence of incomplete information. In [33], the content-based method was integrated with collaborative filtering to overcome the cold start challenge. Regarding the movie cold start, that study collected the additional information from `https://www.imdb.com` in addition to using the rating matrix (advantage). However, like the study reviewed by [24], no acceptable results could be obtained in the absence of sufficient information (disadvantage). In [25], a machine learning approach was proposed by using the k-means algorithms to separate users and develop a neural network for every cluster. The advantage of this approach is 95% accurate in predicting new user ratings but this research only use k-means algorithms and this using that can be one its disadvantage. In [11], a collaborative filtering recommendation method was proposed by integrating the user behavior with feature vectors to enrich the user-item matrix in a hybrid method. The user click rate and purchase history were taken into account to determine the user behavior. Moreover, a nonlinear similarity measure was employed to find the similar users by calculating the Euclidean distance. Euclidean spacing as the similarity criteria are used to discover information. The advantage of this research results indicated higher accuracy of recommendations than the conventional methods but the disadvantage of this research is disregarded the user cold start. In [8], a hybrid method was proposed through collaborative filtering and evolutionary clustering. In the first step, the rating matrix was preprocessed through normalization and dimensionality reduction to obtain denser rating data, which were then employed to implement dynamic evolutionary clustering. Finally, the nearest neighbors with the highest degrees of similar interests were searched for. The matrix dimensionality reduction was used in that study to reduce the problem space complexity, something which might destroy some data. This is one of the disadvantages of the present research. In [3], a movie recommender system was proposed through the k-means and k-nearest algorithms to find similar users and make recommendations to users in the cold start conditions. According to the research results, the proposed method obtained fewer clusters; furthermore, a disadvantage of this algorithm was the negligence of user ratings for items. In [4], a collaborative filtering model was proposed to deal with the new user cold start challenge by using the fuzzy c-means clustering on the user information and proposing a new similarity measurement formula. The results indicated the improvement of recommendations in the absence of sufficient data in the cold start challenge. In [6], the cold start challenge was mitigated by focusing on three main aspects of user information (i.e., common ratings of users, explicit information of reliability, and implicit information of reliability), which were employed to create the target users new rating index. After that, the new similarity measure was defined through the Jaccard similarity index to calculate user similarity. According to the results, the cold start problem mitigated, the disadvantage of this research is that it would be difficult to find explicit and reliable users. In [15], the behavioral data obtained from social media were merged with rating data to make recommendations. After the user information was collected from Twitter, machine learning and decision tree methods were employed to create the user index and make recommendations. The results indicated the alleviation of the cold start and use of reliable data for recommendations; The disadvantage of this result is the incomplete user information in Twitter accounts or the unwillingness of users to complete their account information could affect the quality of recommendations. In [9], the main focus was on the new user cold start challenge by proposing a novel approach to classification of customers through association rules. That study was conducted in two phases: clustering users and making recommendations. The

k-medoids clustering algorithm was employed to group the users. After that, an association rule algorithm was adopted to find the relationships of rated items and make recommendations based on the highest ratings through those algorithms. The mean precision of the proposed solution was similar to those of other algorithms (disadvantage); however, the proposed method was simpler. In [23], a model was proposed to reduce sparseness and deal with the cold start challenge by using an LOD knowledge base to find the information of items in addition to the matrix factorization method for mitigating sparseness. The research results improved the accuracy of recommendations by merging the linked open data with collaborative filtering but the disadvantages are using the additional information of only one web site and disregarding the new user cold start. In [31] to overcome cold start and sparse data challenges, employing contextual similarity measures, utilizing the features of the users and items, and applying machine learning methods have been presented. A method called the context feature singular value decomposition is presented as the first step. In the second step, matrices obtained in the previous step are applied as components of a multi-level singular value decomposition matrix and momentum stochastic gradient descent feature to reduce sparse data. The results demonstrate that the context feature singular value decomposition method helps recommender systems determine which items should be recommended to users via considering preferences and contextual conditions. The evaluation results show that the proposed method outperforms the other approaches in terms of reducing cold start due to four innovative similarity measures of users features similarity, user contextual similarity measures, items similarity measures, and items contextual similarity measures. In [30], proposed a new technique called CSSVD[1] that uses the innovative similarity criteria of item properties, users, and two-level SVD matrix comparison methods as a solution to the cold start problem and apply the contextual information similarity criteria with the help of the tensor property to reduce cases for users who are not ranked, i.e., sparse data. At First, user demographic feature and item property feature measures were used to design the user-item feature matrix, and also the DPCC[2] and IFPCC[3] similarity measure was used to create a user-item similarity matrix to reduce the cold start challenge. Finally, since the resulting matrix of a three-dimensional matrix was used to reduce calculations and higher convergence speed, the tensor property was used with the help of momentum stochastic gradient descent. Experiential results illustrate that the proposed algorithm CSSVD is better than TF, HOSVD[4] , BPR[5] , and CTLSVD in terms of Precision, Recall, F-score, and NDCG measure. Results show the improvement of the recommendations to users through alleviating cold start and sparse data. In [35], a hybrid method was proposed to deal with the cold start challenge. The proposed method consisted of two phases. In the first phase, the similarity values of users were calculated with respect to a hybrid of similarities obtained from the user-item rating matrix and user demographics information. The cosine similarity measure was employed to calculate the similarity values based on the user-item rating matrix. At the same time, the demographic similarities of users were calculated through the weight means of demographic data. Therefore, the final similarity value of users was determined through the linear combination of both consider and demographic similarity criteria. After that, in the second phase of the proposed method, the similarity values of users were calculated through the cosine similarity

[1]Context similarity singular value decomposition
[2]Demographic Pearson Correlation Coefficient
[3]item features the Pearson correlation coefficient
[4]Higher-order singular value decomposition
[5]Bayesian personalized ranking

measure function based on the extended user rating profile to predict the unvisited items for the target users. The results indicated a significant improvement in the proposed method in comparison with the other proposed approaches. However, one of the disadvantages of this research is that it examines the entire data set each time the algorithm is run and does not use a more accurate similarity measure. Table **??** presents an overview of the related works in addition to their advantages and disadvantages and the biggest challenge they have examined.

3. Proposed method

This section presents the proposed method for improving the cold start challenge in recommender systems by using the user demographics information. The proposed method consists of two main phases in which the user demographics information existing in the system will be used with the rating information to improve the cold start challenge. In the first phase, the user-item matrix is created, and clustering is performed then based on the user index information. For this purpose, the user index information is preprocessed. With the arrival of a user with a cold start in the second phase, the appropriate cluster is found. After that, the similarity of cluster users to the target user is calculated through a hybrid similarity measure that is a linear combination of demographic similarity and user rating similarity to determine the target users neighbors. Finally, a list of the best movies is presented to the target user based on the highest predicted rating. Figure 1 demonstrates the block diagram of the proposed method.

The diagram block description is given below:

- Phase 1 : The first step is to form a user-item (movie) matrix, which considers the entire space as a two-dimensional matrix of users and items, and the element of matrix is the value of rating which is rated by users to the movies. This matrix is formed with the ranking information of users who have already done so in the rating system. If this matrix is called $R_{(u,i)}$, the dimensions of this matrix will be u i, so that the rows of the matrix represent the users as $u = u_1, u_2, u_3, ..u_n$, where n is the number of users. Matrix columns represent the items $i = i_1, i_2, i_3, ..., i_m$ where m is the number of items or movies. The rankings in the present data set are based on the numbers 1 to 5, which show 1 disinterested and 5 most interested, which are expressed as the values of the matrix elements. In the second step of the first phase, after pre-processing the user data of the clustering, the clustering is done based on the user profile information, and the proposed clustering algorithms will be explained in Section 2-3.
- Phase 2: A user with cold start is enter. First of all, in step 1 the similarity of the target user with the cluster centers is calculated (based on his/her demographic's information) and select the appropriate cluster. Then in step 2 the hybrid similarity of users (ranking and demographics) in the cluster with target user is calculated and the neighbors for target user are chosen. In step 3 the Selection of the best films based on the combination of GMR and GUC strategies are done and in step 4 Predict rankings and provide recommendations based on the highest ranking to the target user. Finally, the list of predicted films shows to the target user.

3.1 *Data normalization and similarity calculation of users*

This subsection addresses the data normalization method first. After that, the method of calculating similarity of users is presented. Every user has a set of

Table 1. An overview of the related works.

Classification (Based on methods)	Idea	Paper	Advantages	Disadvantages
Using additional information.	Demographic information of social networks.	[31] [30] [35] [19] [13] [33]	-Determining similar users and items will be more accurate -Prediction is based on not only the previous ratings of users but also the characteristics and behavior or users -It can also be used in the absolute cold start state -The characteristics of items are taken into account in addition to item ratings	-It cannot be effective in the absence of additional information of users -It faces the problem of synonymous words in the specifications of items such as humor -It does not record accurate specifications of items
Analyzing the cold start.	Improving the cold start.	[31] [30] [15] [4][13] [33] [8] [3] [9][23]	-improving the user cold start -Improving the item cold start	- It is impossible to accurately determine the user preferences - It disregards the users with particular preferences -It changes the user preferences over time -Similarity of specifications of items -It does not register the accurate information of items
Using the similarity measure.	Proposing the new similarity index or improving the available criteria or hybrid similarity measure.	[31][16] [8]	-Finding the similarities of users more accurately based on the rating matrix -Finding the similarities of items more accurately based on the rating matrix	-Lack of sufficient accuracy in some similarity criteria -Emptiness of the matrix reduces the accuracy of similarity criteria

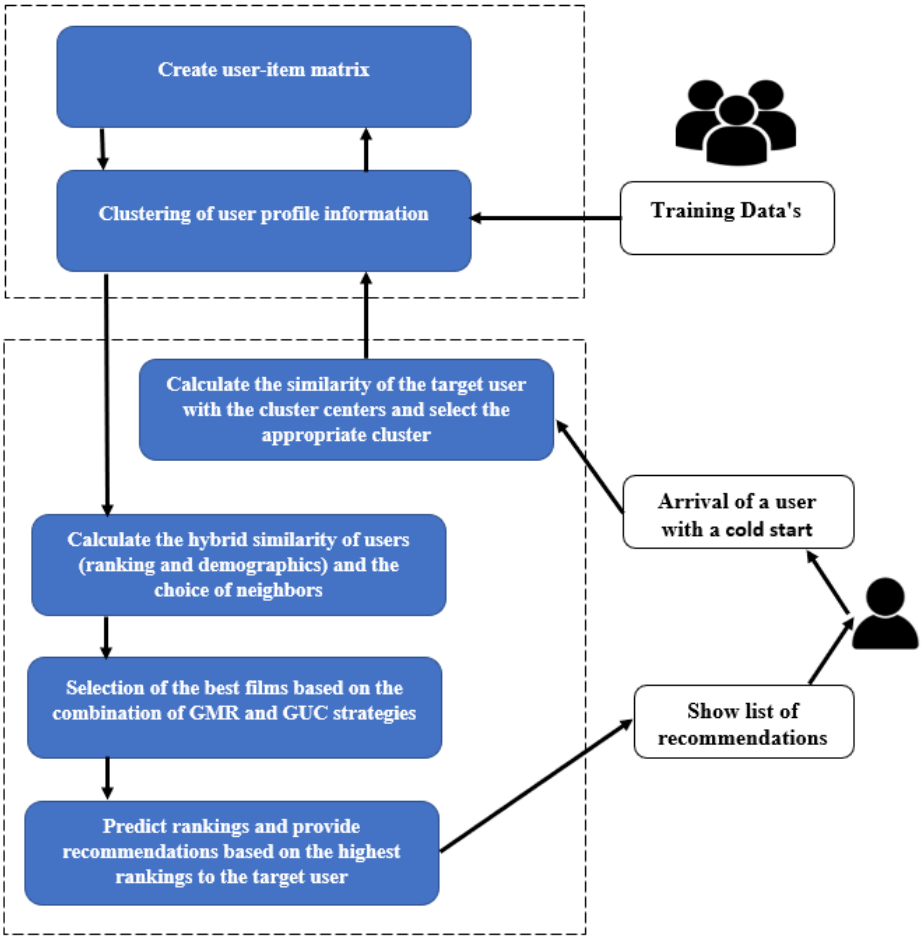

Figure 1. The block diagram of the proposed method.

features constituting the user index. In the proposed method, three attributes were employed to describe the user features. The three attributes were age, gender, and occupation, out of which only age had a numerical value, whereas the other two had categorical values. Since many of the clustering algorithms operate based on distance and are unable to process numerical values, it is first necessary to normal the values of user features. Moreover, similarity and distance are inversely related; therefore, the similarity calculation equation is determined. For this purpose, the method of converting each categorical feature into a numerical one and making them unscaled is presented below:

- Age: The age feature is a numerical value which is defined in the data based as a number based on year for every individual. The variation range of this feature is more extensive that those of others. Therefore, the age normalization approach was employed in the proposed method. In other words, the ages of all individuals in the dataset will be normalized through (1) [14].

$$N_{age_u} = \frac{age_u - age_{min}}{age_{max} - age_{min}} \tag{1}$$

Where age_u denotes the age of user u, and age_{min} refers to the minimum age in the dataset, whereas age_{max} indicates the maximum age in the dataset. The above equation helps define the ages of individuals as normal values N_{age_u}

Table 2. The categorization of occupations [5].

Row	Title
1	Administrator, executive, marketing, salesman
2	Doctor, healthcare
3	Homemaker, retired
4	None, other
5	Scientist, programmer, engineer, technician
6	Educator, librarian, lawyer
7	Entertainment artist, writer
8	Student

within [0, 1]. After normalization, this equation can be employed to map age_{min} and age_{max} onto 0 and 1, respectively. In fact, age is a numerical continuous datum, whereas the Euclidean distance is among the most widely-used criteria for calculating the distance between numerical and continuous features [14]. (2) indicates the formula employed to calculate the similarity of ages for two users based on the Euclidean distance.

$$sim(N_{age_u}, N_{age_v}) = 1 - \sqrt{\mid N_{age_u} - N_{age_v} \mid^2} \tag{2}$$

Where N_{age_u} and N_{age_v} are the normalized ages of users u and v, respectively, and $sim(N_{age_u},N_{age_v})$ denotes the similarity of two users based on the age feature. The smaller this value, the more similar the two users, and vice versa.

- Gender: For every individual, this feature can be either M (for male) or F (for female). Therefore, this feature is converted into numerical values by placing 1 and 0 if the user selects male or female, respectively. (3) is employed to calculate this feature.

$$sim_{gender}(u, v) = \begin{cases} 1, if\ gender_u = gender_v \\ 0, if\ gender_u \neq gender_v \end{cases} \tag{3}$$

Where $sim_{gender}(u, v)$ indicates the similarity between u and v in terms of gender. If two users have the same gender, the similarity value is 1; otherwise, it is 0.

- Occupation: To convert the feature of occupation into numerical values, a unique list of all occupations was prepared with respect to all users existing in the database. In MovieLens, 21 different occupations were defined, and every occupation was given a corresponding number between 0 or 1 so that this feature could be placed in a similar interval to age and gender. To simplify the calculation of occupation similarity between users, the occupations were first categorized. The occupations with higher similarities were placed in one group. Table **??** indicates the categorization of occupations [5].

  If the occupations of two users are completely similar, the numerical value is considered 1. If their occupations are classified as the same category, the value is 0.5; finally, if their occupations are classified as different categories, the value is considered 0. (4) defines this process:

$$sim_{job}(u, v) = \begin{cases} 1, if\ j_u = j_v \\ 0.5, if\ j_u \neq j_v\ and\ C(j_u) = C(j_v) \\ 0, if\ j_u = j_v\ and\ C(j_u) \neq C(j_v) \end{cases} \tag{4}$$

Where j denotes the occupation feature, and j_u indicates the value of this feature for u, whereas $C(j_u)$ refers to the occupation category of user u. In this equation, $sim_{job}(u, v)$ indicates the similarity between u and v inters of occupation.

A numerical matrix is obtained from the user index information preprocessing to describe the user index. In this matrix, every row indicates a user, whereas every column represents the features of a user. Finally, (5) is employed to calculate the similarity of two users in terms of demographic features [35].

$$sim(u, v) = \frac{\sum_{j=1}^{n} s_j w_j}{\sum_{j=1}^{n} w_j} \tag{5}$$

Where $sim(u, v)$ denotes the similarity between u and v, and s_j indicates the value of similarity in every feature. Furthermore, w_j and n refer to the corresponding weight of every feature and the number of features, respectively. For the corresponding weights (w_j) in this article different scenarios Table **??** have been tested.

3.2 *Users clustering*

Following the first phase in the proposed method, users clustering is performed with respect to the users profile matrix, which was introduced in Subsection 3.1, by using the k-means, k-medoids, and k-prototype clustering algorithms. These algorithms operate based on the distances of samples. (6) is employed to calculate the value of this distance [14].

$$dist(u, v) = 1 - sim(u, v) \tag{6}$$

The value of $sim(u, v)$ is obtained from (5). After clustering, the users with the highest similarity are placed in the same cluster, whereas the users with the lowest similarity are placed in different clusters.

3.3 *Arrival of user with cold start and discovery of neighbors*

In the second phase of the proposed method, the target user enters the system. His/her demographic features are calculated in the same way as other users in order to determine his/her appropriate cluster by calculating the distance between the target user and cluster centers. After the target users appropriate cluster is selected, his/her neighbors are determined through the hybrid similarity measure consisting of their rating similarity and demographic similarity. For this purpose, (7) is used [18]:

$$sim(u, v) = \alpha \times sim_{demo} + (1 - \alpha) sim_{apcc} \tag{7}$$

Where sim_{demo} is obtained from (6), and sim_{apcc} refers to the rating similarity of users within the selected cluster based on the asymmetric Pearsons correlation. Moreover, α is a parameter that indicates the dependency of final similarity on each of the demographic and rating similarities. In this equation, $\alpha = 1$ represents the complete dependency of final similarity on demographic information. So this equation can cover both modes (user with absolute cold start and relative cold

Table 3. An example of user ratings for items [27].

	Item1	Item2	Item3	Item4	Item5	Item6
User1	4		2			
User2	4	1	2	1	1	1
User3		5		3		
User4		1	2			
User5	4				5	5
User6	1		2		1	1

start. An appropriate similarity measure must be employed to calculate the rating similarity. The Pearson correlation coefficient (PCC) is among the most common criteria. (8) is utilized to calculate the PCC:

$$P(u,v) = \frac{\sum_{i=1}^{n}(r_{ui} - \overline{r_u})(r_{vi} - \overline{r_v})}{\sqrt{\sum_{i=1}^{n}(r_{ui} - \overline{r_u})^2}\sqrt{\sum_{i=1}^{n}(r_{vi} - \overline{r_v})^2}} \tag{8}$$

Where r_{ui} and r_{vi} are the user ratings of u and v for the ith movie, and $\overline{r_u}$ and $\overline{r_v}$ are the means of user ratings of u and v, respectively. Moreover, n refers to the number of items rated by both users. Although the PCC proved to be successful in many studies, there were a few problems, the most important of which is that $sim(u,v)$=$sim(v,u)$ [27]. For instance, Table **??** presents a case of user ratings: According to Table **??**, User1s ratings are completely similar to User2s ratings; however, the reverse is not true. Conventional methods like the PCC cannot distinguish between these two users with different rating characteristics. In fact, they consider the ratings of these two users similar; hence, the effect received by User1 from User2 in predicting the rating of a new movie is equal to the effect received by User 2 from User 1 in recommending a new movie. To prevent this contradiction, an asymmetric similarity measure was used in [26]. According to (9), this measure includes the ratio of items with common user ratings to the items rated by the target user.

$$sim(u,v) = \frac{|I_u \bigcap I_v|}{|I_u|} \tag{9}$$

Where I_u and I_v refer to the sets of items rated by u and v, respectively. However, this equation considers only the ratio of items rated by the target user to the other users but disregards the ratio of rating among all users. Therefore, another parameter is also added to this equation, which is then rewritten as (10) [26].

$$sim(u,v) = \frac{|I_u \bigcap I_v|}{|I_u|} \times \frac{2 * (I_u \bigcap I_v)}{|I_u| + |I_v|} \tag{10}$$

According to (10) defined as the ratio of items with common ratings between users, it is normalized by the number of items rated by the active users. This equation can also be expressed as (11) [27].

$$sim(u,v) = \frac{2 * (I_u \bigcap I_v)^2}{|I_u| * (|I_u| + |I_v|)} \tag{11}$$

Based on (11) , the similarity measure considers only the number of common ratings between two users. This equation can be used as a coefficient with the other similarity criteria. In the proposed method, this measure is used as a coefficient for the Pearsons similarity measure in (8) to create a new similarity measure called the APCC. After the rating similarities of users were calculated through this similarity measure called the asymmetry Pearsons similarity measure, a matrix of user similarity is obtained from the user ratings for movies. The resultant values of this matrix are used in (7).

3.4 *Rating prediction*

After the target users neighbors are selected, a list of the top movies is obtained from the ratings of the users neighbors for rating prediction. For this purpose, GMR and GUC strategies mentioned in [35] are combined. In the first strategy, the number of ratings given by the neighbors set of the target user to each movie is calculated first. After that, the movies are sorted with respect to the number of ratings and the highest rating mean. A few of them are then selected as the top movies. In the second strategies, the list of top movies obtained from GMR and the list of movies selected by the user index are employed to calculate the similarity of each to the other movies existing in the dataset. These movies are then sorted downward based on their similarities to the resultant movies. A number of I movies are then recommended to the target user. (12) is employed to calculate the similarities of movies [35]. (12) [14].

$$sim(i,j) = \frac{\sum\limits_{n \in N} r_{n,i} r_{n,j}}{|N(u)|} \tag{12}$$

Where $|N(u)|$ denotes the target users set of neighbors, and $r_{n,i}$ refers to the nth users rating for the ith movie. Finally, (13) is employed to predict the target users rating for the designated movies which are not rated (not visited) by the target user [35].

$$pre(u,i) = \frac{\sum\limits_{n \in N} sim(u,v) * r_{vi}}{\sum\limits_{n \in N} sim(u,v)} \tag{13}$$

Where $pre(u,i)$ represents the predicted rating of u for the ith movie, and r_{vi} is the rating given by the neighboring user to the ith movie. (7) is employed to calculate $sim(u,v)$.

4. Evaluation of results

In this section, several experiments are conducted to evaluate the performance of the proposed method. The evaluation results are then compared with the results of the methods introduced in [35]. The proposed method was implemented in MATLAB 2018b. The corresponding code was executed on a system equipped

with Intel Core i7 @4.2 GHz and 8 GB of RAM. The following subsections present the research dataset, evaluation criteria, and experimental results in detail.

4.1 *Dataset*

In this study, the MovieLens[1] dataset was employed to evaluate the proposed method. Containing the real data, this standard dataset was used in different studies ([35], [36], [37]). MovieLens 100k is the version used in this study. It contains 10000 ratings from 1 to 5 given by 943 users to 1682 movies. In this dataset, the users did not rate all movies. In fact, they rated only 20 movies at least.

4.2 *Evaluation metrics*

The dataset was first divided randomly into training and test sections to evaluate the proposed method. In other words, 80% and 20% of data were allocated to training and test sections, respectively. The training section was used in the first phase to develop the target user model, whereas the test section was used as the cold start users in the second phase. The real ratings of users given by the test users to the movies, and these users were entered the system as the cold start users to determine their predicted ratings through the proposed method. After that, the predicted ratings were compared with the real ratings to determine the rates of precision and error in the proposed method. The following evaluation criteria were employed to calculate precision and error.

4.2.1 Mean absolute error (MAE)

This metric measures the mean absolute error of the predicted rating to calculate the precision of the recommender system accurately. For this purpose, the absolute value of difference between the predicted error from the real rating is measured, and the total mean error is considered the absolute error. (14) indicates the mean absolute error ([31], [35]]).

$$MAE = \frac{\sum_{u}\sum_{i}|p_{u,i} - r_{u,i}|}{n} \tag{14}$$

Where $p_{u,i}$, $r_{u,i}$ and n denote the predicted rating, real rating, and total number of predictions.

4.2.2 Rating coverage (RC)

This metric measures the ratio of items that the recommender system can offer. In fact, RC predicts the ratio of the predicted test movies to the total movies. The larger this value, the better the coverage rate of the recommender system. (15) indicates how to calculate this metric [35]:

$$RC = \frac{n}{m} \tag{15}$$

Where n and m represent the number of predicted ratings and the total number of ratings.

[1] http://grouplens.org/datasets/movielens/

Table 4. Descriptions of parameters used in the proposed method.

Parameters	Notation	Notation
Initial values rated in the target user profile	N	0,2,3,4,5,10
Number of movies recommended	I	5,10,15,20,50
Number of neighbors	K	25
Weight value of the users age attribute	w1	0.3
Weight value of the users gender attribute	w2	0.1
Weight value of the users occupation attribute	w3	0.6
Weight value of the users occupation attribute	C NO	K-MEANS C No=8 K-MEDOIDS C No=8 K-PROTOTYPES C No=9

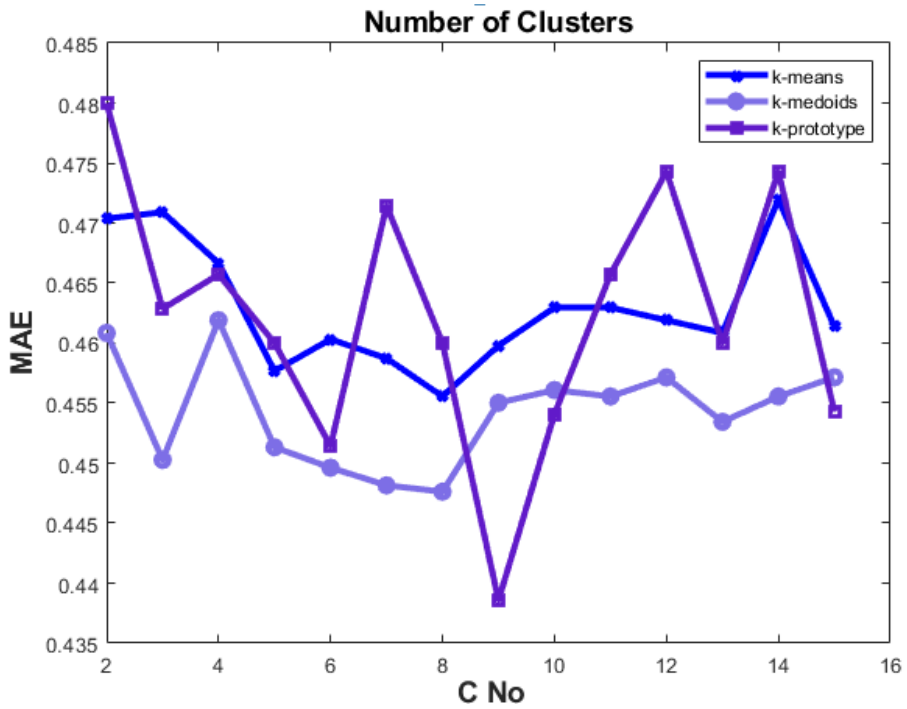

Figure 2. Determining the optimal number of clusters based on the number of clusters and MAE.

4.3 *Results*

In the first phase of the proposed method, the first step is to create the user-movie matrix containing the ratings given by users to the movies. In this matrix, each row represents a user, whereas each column represents the movies. The elements of this matrix indicate the ratings given by every user to the movies (from 1 to 5). In the second step of the first phase, it is necessary to perform clustering based on the user demographics information after data normalization. For this purpose, the parameters of the proposed method should first be determined for evaluation. Table **??** describes the parameters: It should be mentioned that the values of N and I have been chosen randomly (Exactly the same as the numbers used in [35]), so that the results of the proposed method can be compared with the results of research [35]. To determine the optimal number of clusters for every algorithm, calculations are based on the MAE and the number of clusters. Figure 2 reports the results:

For this purpose, the trial-and-error technique was adopted. In other words, the proposed method was executed with the initially set values for different numbers

of clusters. Every cluster number with the lowest error rate is considered the optimal number of clusters. According to Figure 2, the X-axis indicates the number of clusters, whereas the Y-axis represents the MAE for every cluster. The test process started with the initial value of 2 clusters for all three algorithms, and the total MAE was calculated. This process continued in the same way as the number of clusters increased. According to the results, there were 8 clusters with the lowest MAE in the k-means and k-medodis algorithms, whereas there were 9 such clusters in the k-prototypes algorithm. Based on the resultant number of clusters, the clustering operations were then performed.

In the evaluations of the second phase in the proposed method, calculations are performed then based on different values of N and I. The first parameter (N) indicates the number of movies rated by the target user. For this purpose, two states are taken into account. In the first state, the target user is assumed to be in the cold start mode; therefore, N is considered zero. In the second state based on the method proposed by [35], a few movies are randomly considered the initial values of ratings given by the target user (N= 2, 3, 4, 5, 10) to perform the calculations. Finally, the results are compared with those reported by [35]. Figures 3 and 4 indicate the results obtained from the first state, in which the target user has no initial rating, based on the MAE in comparison with the method proposed by [35] for I= 10.

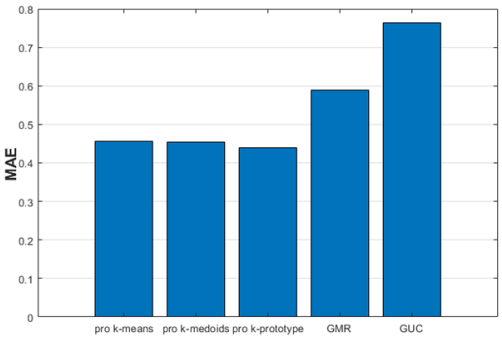

Figure 3. Comparison of the proposed method in terms of N=0 and the MAE.

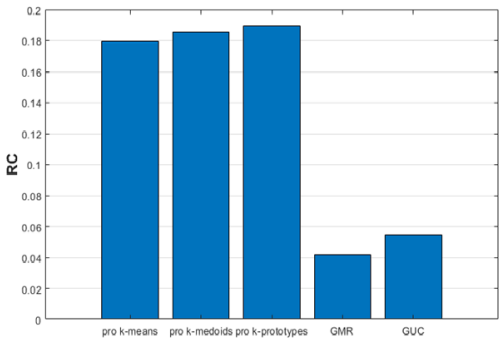

Figure 4. Comparison of the proposed method in terms of N=0 and the RC.

In Figures 3 and 4, the X-axis indicates the proposed method with all three clustering algorithms as well as GRM and GUC strategies in [35], whereas the Y-axis represents the MAE and RC for each of the executed methods.

According to Figure 3, the MAE was reported 0.4571, 0.4550, and 0.4402 in the proposed method for the k-means, k-medoids, and k-prototypes algorithms, respectively. These values are lower than the MAE rates of GRM and GUC strategies (i.e., 0.5896 and 0.7634, respectively). Since the MAE shows the difference between the predicted values and the real values, a lower MAE indicates the more accurate prediction of movies than the method proposed by [35]. Therefore, the proposed method in this study yielded a lower error rate and more accurate results for a cold start user with no initial ratings than the method introduced by [35]. According to Figure 4, the RC rates of the proposed method were 0.1796, and 0.1859, and 0.1892 for the three algorithms, respectively, whereas they were reported 0.0419 and 0.549 for the methods proposed by [35], respectively. It can be concluded that the proposed method found a larger number of movies for prediction than the method introduced by [35] and also performed better in make diverse recommendations. The comparison of results from the three clustering algorithms indicated that the k-prototypes clustering algorithm had a lower MAE rate when the tar-

get user had no initial ratings. Hence, the proposed method with the k-prototypes algorithm yielded better results when the dataset had more non-numerical characteristics than numerical characteristics. After the k-prototypes algorithm, the k-medoids algorithm with the difference function defined as (6) had a lower MAE and a higher RC. Hence, it can be inferred that the defined difference function managed to reduce the MAE and improve the system performance significantly. However, in general, the comparison of all three clustering algorithms indicated that they had slight differences, something which depicted that all three clustering algorithms outperformed the method introduced by [35] and using no clustering techniques in terms of MAE reduction and RC improvement.

In the second state, the target user is assumed to have initial ratings. However, since there are a small number of ratings, it is impossible to determine the user preferences accurately; thus, the user is considered a cold start user. To evaluate this state, a few movies are selected randomly as the target users preferences. After that, the proposed method was executed for different values of N and I=10, and the results were compared in terms of MAE and RC. Figures 4 and 5 report the results based on MAE and RC.

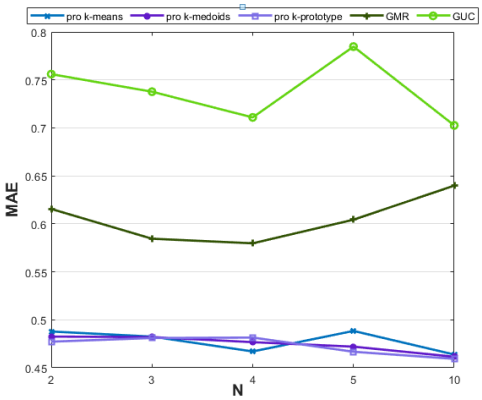

Figure 5. Comparison of the proposed method in different values of N and MAE.

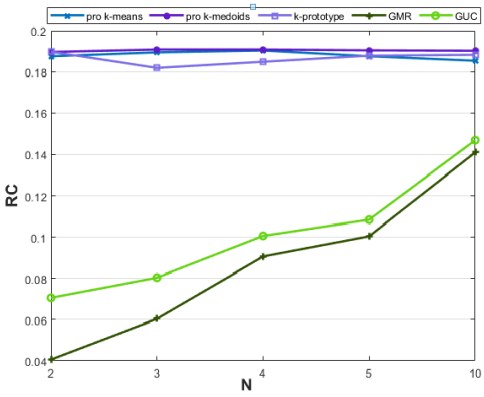

Figure 6. Comparison of the proposed method in different values of N and RC.

In Figures 5 and 6, the X-axis represents the number of initial recommendations, whereas the Y-axis indicates the values of MAE and RC. According to Figure 5, the proposed method yielded the lowest MAE for the k-prototypes clustering algorithm than the other proposed methods. The GMR strategy outperformed the GUC strategy; however, it yielded a higher MAE than the proposed method for all three clustering algorithms. This finding indicates that the combination of GMR and GUC and use of clustering algorithms through the hybrid similarity measure for different values of N managed to make recommendations with lower MAE than the method introduced by [35], something which proved the higher accuracy of the proposed method in the prediction of movies. In addition, Figure 5 demonstrates that increasing the initial ratings given by users in the three proposed algorithms, the MAE decreased; therefore, the better the target users preferences are identified, the more accurate recommendations are presented. According to Figure 6, the proposed method managed to yield a better RC than the similar methods. In other words, it predicted a larger number of test movies than the total movies in comparison with the method proposed by [35] and succeeded in predicting more diverse movies. However, the RC remained nearly constant for all three algorithms in the proposed method, although it was larger than the other similar method. This could explain that the users of a cluster liked a limited number of movies because they have similar behavior and receive similar recommendations. However,

Table 5. Comparing the proposed method with different clustering algorithms in MAE and RC based on N=3.

	Metrics	GMR	GUC	Proposed K-means	Proposed K-medoids	Proposed K-prototypes
I=5	MAE	0.6995	0.8911	0.5386	0.5545	0.5566
	RC	0.0548	0.0518	0.1891	0.1883	0.1977
I=10	MAE	0.5946	0.8127	0.4910	0.4926	0.4810
	RC	0.0549	0.0529	0.1903	0.1896	0.1818
I=15	MAE	0.6366	0.7544	0.4451	0.4508	0.4384
	RC	0.0546	0.0544	0.1885	0.1892	0.1875
I=20	MAE	0.7860	0.7688	0.4259	0.4299	0.4230
	RC	0.0546	0.0546	0.1881	0.1889	0.1849
I=50	MAE	0.7415	0.7424	0.3331	0.3396	0.3399
	RC	0.0562	0.0542	0.1873	0.1895	0.1839

in the two methods proposed by [35], since the entire data set was employed to find similar movies, increasing the value of N increased the RC. Hence, the higher the users initial rating, the more diverse the recommended movies. As a result, it is superior to the proposed method. However, it can generally be concluded that the proposed method was more accurate than the similar method when the target users initial ratings were low. Thus, the combination of GMR and GUC and use of clustering as well as the hybrid similarity measure proposed a better method when the initial ratings were low. Different evaluation results are analyzed in terms of MAE and RC for N=3 and different values of I. Table **??** repots the evaluation results. In Table **??**, the columns depict the names of methods selected for comparison, whereas the rows indicate the number of recommendations. According to Table **??**, the proposed method yielded a lower MAE (and more accurate predictions) for the three clustering algorithms in all cases; thus, it outperformed the method introduced by [35]. For instance, the MAE of GMR was reported 0.5946 for I=10; thus, GMR showed the best performance and lowest MAE for different values of I. However, the three clustering algorithms in the proposed method yielded 0.4910, 0.4926, and 0.4810, which indicate the lowest MAE and the best performance. In the proposed method, increasing the number of initial recommendations decreased the MAE in all three clustering algorithms. In other words, the larger the number of recommendations offered to the target user, the higher the chances that the target user likes the recommendations. All the three algorithms in the proposed method yielded larger values of RC than GMR and GUC; however, the value of RC remained constant as I increased. In other words, the proposed method made more diverse recommendations in general; however, the value of RC in [35] were different for various values of I. According to Table **??**, increasing the value of I improved the value of RC in GUC. In fact, the larger the number of the recommended movies, the more easily the similar movies are found in this method. This is an advantage over the proposed method. The other experiments conducted on the proposed method indicated that similar results were obtained for the other values of N. It can generally be concluded that increasing the number of recommendations to the target user helped the proposed method outperform the method introduced by [35] in terms of MAE (accuracy of predictions) and RC (diversity of recommendations). Therefore, the proposed method improved to some extent. However, although it yielded better values of RC, these values remained constant

Table 6. Different weighting scenarios of demographic characteristics [31]

Scenarios	Weights
Scenarios1	w1=0.33, w2=0.34, w3=0.33
Scenarios2	w1=0.6, w2=0.3, w3=0.1
Scenarios3	w1=0.3, w2=0.6, w3=0.1
Scenarios4	w1=0.3, w2=0.1, w3=0.6

in all different executions. This is due to the similarity of user preferences in every cluster.

4.4 *Sensitivity analysis of the proposed method to different parameters)*

This subsection analyzes the sensitivity of the proposed method to the values of α and w. The first experiment measures the effect of α. Defined as a real number ranging from 0 to 1, this parameter indicates the effects of demographic similarity and user rating similarity in (7) . Therefore, the larger this parameter, the greater the effects of demographic characteristics. Figures 7 and 8 demonstrate the effects of α on MAE for the values of I=10 and N=0.

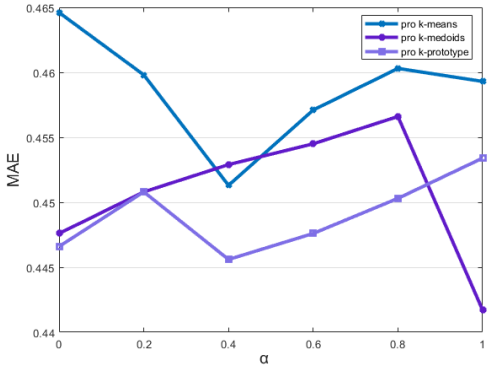

Figure 7. The effect of α on MAE.

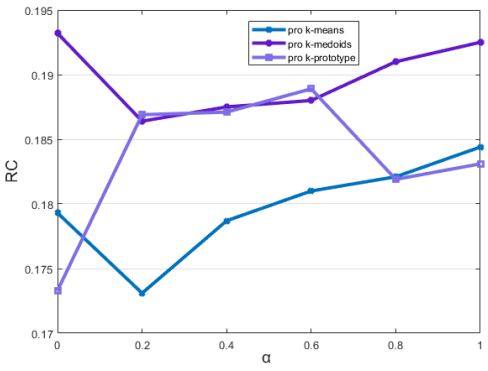

Figure 8. The effect of α on RC.

In Figures 7 and 8, the X-axis shows different values of α, whereas the Y-axis depicts the values of MAE and RC. According to the evaluation results in Figure 7, the proposed method yielded the best accuracy and the lowest MAE for $\alpha = 0.4$ in the k-means and k-prototypes algorithms and for $\alpha = 1.0$ in the k-medoids algorithm. Moreover, the evaluation results in Figure 8) demonstrated good values of RC for all three algorithms when $\alpha = 1$, i.e., the recommendations are based only on the user demographics information . In this case, the proposed algorithm managed to make more diverse recommendations. However, the k-medoids and the k-prototypes algorithms yielded larger values of RC for different values of α. Determining the accurate value of α depends on the type of business and its goal.

The next parameter is the value of w that indicates the weights of demographic characteristics (i.e., age, gender, and occupation). Table **??** reports different weights of demographic data [35]. It should be mentioned that $w_j \in [0, 1]$ and $\sum_{j=1}^{3} w_j = 1$.

Figures 9 and 10 demonstrate the evaluation results based on different values of demographic characteristics.

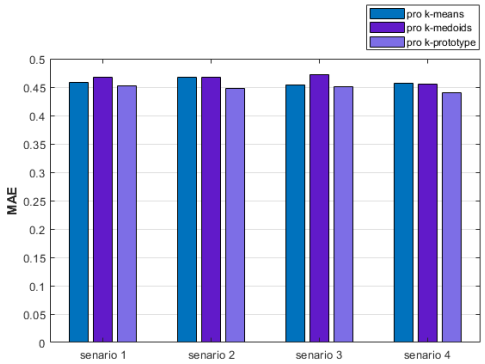

Figure 9. The weighting scenarios of demographics information based on MAE.

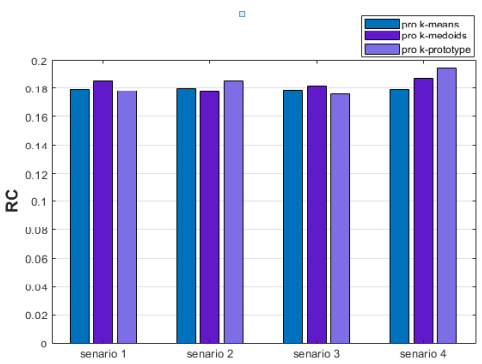

Figure 10. The weighting scenarios of demographics information based on RC.

According to Figures 9 and 10, the weights of user demographics information affected what features the users were considered similar as well as the accuracy and coverage of the selected ratings. The results indicated that the k-prototypes algorithm yielded a smaller MAE in the fourth scenario, whereas the k-means in the third scenario and the k-medoids in the fourth scenario had lower MAEs. Moreover, the k-prototypes, k-medoids, and k-prototypes algorithms yielded the largest values of RC in the fourth, first, and second scenarios. Hence, it can be concluded that selecting the weights of demographics information can affect the prediction error. The accurate weights are determined with respect to the business goals and conditions of users.

5. Conclusion and future suggestions

This paper proposed a novel method for improving a major challenge in recommender systems called the cold start challenge. After information preprocessing in the proposed method, the users are clustered with respect to their demographics information (i.e., age, gender, and occupation). The clustering process was performed through the k-means, k-medoids, and k-prototypes algorithms. The target users neighbors were then selected by using a hybrid similarity measure consisting of the demographics information similarity and user rating similarity in every cluster through the asymmetric Pearson correlation coefficientis. The combination of these similarity measure helped select stronger neighbors who were more similar to the target user. The recommendations were also made by combining GMR and GUC, and a number of top movies were recommended to the user. According to the results, clustering methods were efficient in reducing the problem space complexity in finding the users who would behave like the target user, thereby improving the system performance. Moreover, the use of a hybrid similarity measure and especially the asymmetric Pearson correlation coefficientis helped find stronger neighbors who were more similar to the target user. According to the results obtained in Figure 3, the mean absolute error of the proposed method with each of the clustering algorithms k-means, k-medoids and k-prototypes is equal to 0.4571, 0.4550 and 0.4403, respectively Compared to the GMR and GUC methods, which have values of 0.5896 and 0.7634, has been able to provide more accurate recommendations in the conditions of absolute cold start, so that the proposed method

compared to the GMR method up to about 26% and the GUC method up to about 34% improved the accuracy of suggestions. The results in Figure 4 shows the same results as in the previous case when the target user has a number of initial rankings. In terms of rating coverage, as shown in the results of Figures 4 and 6, it shows about 60% improvement over the same method and in the same conditions, However, as the results of Figure 6 shows, the coverage of the proposed method remains almost constant with increasing number of initial ranking of target users, while this value for the two methods presented in research [35] increases with the number of initial ranking of users.

In addition, integrating GMR with GUC resulted in the significant MAE reduction as opposed to the method proposed in [35] in addition to more accurate recommendations. These two strategies can also be integrated to make more diverse recommendations for the target user, for the recommendations not only are limited to the preferences of neighboring users but can also help make recommendations similar to the ones visited less often by the neighboring users.

The advantages of the proposed method include using the clustering technique for reducing the problem space complexity, employing the user demographics information to detect the behavioral patterns of users for movies, clustering numerical and categorical features in the dataset, adopting the hybrid similarity measure to select stronger neighbors through the asymmetric Pearsons correlation, integrating GMR and GUC to make accurate and diverse recommendations, and improving the predictions made for users in both absolute and relative user cold start modes. The disadvantages of the proposed method include the random selection of cluster centers in the clustering algorithms and the constant value of RC for the proposed method in different applications. Eventually, it can be concluded that demographic information can be utilized to accurately identify the behavioral patterns of users who enter the system with no prior ratings. If it is possible to obtained efficient similarities between the target user and the other users, it will be possible to predict what preferences the target user may have and what ratings the target user may give to the movies in the future. If the user preferences are identified correctly with an accurate list of recommendations offered to the target user for his/her satisfaction, the users and clients of the target website will escalate, something which will finally lead to profitability. Regarding the area of movie recommendation, this study was limited to the data existing in MovieLens. In future studies, the proposed method can be implemented in other datasets and areas such as music recommendation, book recommendation, and online shopping. Moreover, the future studies can benefit from other clustering methods such as the density-based methods in addition to integrating movie features such as genre and names of directors to improve the problem of constant RC in predictions. It is also possible to address the effect of cold start on social networks, recommendation of items, and even introduction of famous people to one another.

References

[1] C. C. Aggarwal and C. K. Reddy, Algorithms and applications, Chapman&Hall/CRC Data Mining and Knowledge Discovery Series, CRC Press, Londra, (2014).

[2] M. Ahmed, M. T. Imtiaz and R. Khan, Movie recommendation system using clustering and pattern recognition network, 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, (2018) 143–147.

[3] R. Ahuja, A. Solanki and A. Nayyar, Movie recommender system using K-means clustering and K-nearest neighbor, 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, (2019) 263–268.

[4] N. F. AL-Bakri and S. Hassan, A Proposed Model to Solve Cold Start Problem using Fuzzy User-Based Clustering, 2019 2nd Scientific Conference of Computer Sciences (SCCS), Baghdad, Iraq, (2019) 121–125.

[5] M. Y. H. Al-Shamri, User profiling approaches for demographic recommender systems, Knowledge-Based Systems, **100** (2016) 175–187.
[6] M. Ayub, M. A. Ghazanfar, Z. Mehmood, K. H. Alyoubi and A. S. Alfakeeh, Unifying user similarity and social trust to generate powerful recommendations for smart cities using collaborating filtering-based recommender systems, Soft Computing, **24 (15)** (2020) 11071–11094.
[7] R. Chen, Q. Hua, Y. S. Chang, B. Wang, L. Zhang and X. Kong, A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks, IEEE Access, **6** (2018) 64301–64320.
[8] J. Chen, C. Zhao and L. Chen, Collaborative filtering recommendation algorithm based on user correlation and evolutionary clustering, Complex & Intelligent Systems, **6 (1)** (2020) 147–156.
[9] L. C. de Carvalho, F. Rodrigues and P. Oliveira, A hybrid recommendation algorithm to address the cold start problem, International Conference on Hybrid Intelligent Systems, Cham, (2018) 260–271.
[10] S. B. U. Duja, B. Niu, B. Ahmed, M. U. F. Alvi, M. Amjad, U. Ali, Z. U. Rehman and W. Hussain, A proposed method to solve cold start problem using fuzzy user-based clustering, International Journal of Advanced Computer Science and Applications, **11 (2)** (2020) 529–536.
[11] L. Feng, Q. Zhao and C. Zhou, Improving performances of Top-N recommendations with co-clustering method, Expert Systems with Applications, **143** (2020) 113078.
[12] X. Guo, S. C. Yin, Y. W. Zhang, W. Li and Q. He, Cold start recommendation based on attribute-fused singular value decomposition, IEEE Access, **7** (2019) 11349–11359.
[13] G. Guo, J. Zhang and D. Thalmann, Merging trust in collaborative filtering to alleviate data sparsity and cold start, Knowledge-Based Systems, **57** (2014) 57–68.
[14] J. Han, J. Pei, M. Kamber, Data Mining: Concepts and Techniques, The Morgan Kaufmann Series in Data Management Systems, Elsevier, (2011).
[15] J. Herce-Zelaya, C. Porcel, J. Bernab-Moreno, A. Tejeda-Lorente and E. Herrera-Viedma, New technique to alleviate the cold start problem in recommender systems using information from social media and random decision forests, Information Sciences, **536** (2020) 156–170.
[16] S. Jain, A. Grover, P. S. Thakur and S. K. Choudhary, Trends, problems and solutions of recommender system, International Conference on Computing, Communication & Automation, Greater Noida, India, (2015) 955–958.
[17] S. Khusro, Z. Ali and I. Ullah, Recommender systems: issues, challenges, and research opportunities, In: K. Kim and N. Joukov (eds), Information Science and Applications (ICISA), Lecture Notes in Electrical Engineering, Springer, Singapore, **376** (2016) 1179–1189.
[18] Z. Li and L. Zhang, Subspace ensemble-based neighbor user searching for neighborhood-based collaborative filtering, In: G. Li, J. Yang, J. Gama, J. Natwichai and Y. Tong (eds), Database Systems for Advanced Applications, Lecture Notes in Computer Science, Springer, Cham, **11447** (2019) 449–463.
[19] B. Lika, K. Kolomvatsos and S. Hadjiefthymiades, Facing the cold start problem in recommender systems, Expert Systems with Applications, **41 (4)** (2014) 2065–2073.
[20] V. Maihami, D. Zand and K. Naderi, Proposing a novel method for improving the performance of collaborative filtering systems regarding the priority of similar users, Physica A: Statistical Mechanics and its Applications, **536** (2019) 121021.
[21] M. H. Mohamed, M. H. Khafagy and M. H. Ibrahim, Recommender systems challenges and solutions survey, International Conference on Innovative Trends in Computer Engineering (ITCE), (2019) 149–155.
[22] A. Nair and R. Mathews, Challenges and solutions in recommender systems, International Conference on Computer Networks, Big data and IoT, Springer, Cham, (2019) 188–194.
[23] S. Natarajan,S. Vairavasundaram, S. Natarajan and A. H. Gandomi, Resolving data sparsity and cold start problem in collaborative filtering recommender system using linked open data, Expert Systems with Applications, **149** (2020) 113248.
[24] J. Niu, L. Wang, X. Liu and S. Yu , FUIR: Fusing user and item information to deal with data sparsity by using side information in recommendation systems, Journal of Network and Computer Applications, **70** (2016) 41–50.
[25] B. Patel, P. Desai and U. Panchal, 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, (2017) 1–4.
[26] P. Pirasteh, D. Hwang and J. J. Jung, Exploiting matrix factorization to asymmetric user similarities in recommendation systems, Knowledge-Based Systems, **83** (2015) 51–57.
[27] P. Pirasteh, J. J. Jung and D. Hwang, An asymmetric weighting schema for collaborative filtering, In: D. Camacho, S. W. Kim and B. Trawiski (eds), New Trends in Computational Collective Intelligence, Studies in Computational Intelligence, Springer, Cham, **572** (2015) 77–82.
[28] S. K. Raghuwanshi and R. K. Pateriya, Recommendation systems: techniques, challenges, application, and evaluation, In: J. Bansal, K. Das, A. Nagar, K. Deep and A. Ojha (eds), Soft Computing for Problem Solving, Advances in Intelligent Systems and Computing, Springer, Singapore, **817** (2019) 151–164.
[29] F. Ricci, L. Rokach and B. Shapira, Recommender systems: introduction and challenges, In: F. Ricci, L. Rokach and B. Shapira (eds), Recommender Systems Handbook, Springer, Boston, MA, (2015) 1–34.
[30] K. V. Rodpysh, S. J. Mirabedini and T. Banirostam, Employing singular value decomposition and similarity criteria for alleviating cold start and sparse data in context-aware recommender systems, Electronic Commerce Research, (2021), doi:10.1007/s10660-021-09488-7.
[31] K. V. Rodpysh, S. J. Mirabedini and T. Banirostam, Resolving cold start and sparse data challenge in recommender systems using multi-level singular value decomposition, Computers & Electrical Engineering, **94** (2021) 107361.
[32] L. Safoury and A. Salah, Exploiting user demographic attributes for solving cold-start problem in recommender system, Lecture Notes on Software Engineering, **1 (3)** (2013) 303–307.
[33] A. Sattar, M. Ghazanfar and M. Iqbal, Building accurate and practical recommender system algo-

rithms using machine learning classifier and collaborative filtering, Arabian Journal for Science & Engineering, **42 (8)** (2017) 3229–3247.

[34] R. Sharma, D. Gopalani and Y. Meena, Collaborative filtering-based recommender system: Approaches and research challenges, 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT), Ghaziabad, India, (2017) 1–6.

[35] F. Tahmasebi, M. Meghdadi, S. Ahmadian and K. Valiallahi, A hybrid recommendation system based on profile expansion technique to alleviate cold start problem, Multimedia Tools and Applications, **80 (2)** (2021) 2339–2354.

[36] B. Walek and V. Fojtik, A hybrid recommender system for recommending relevant movies using an expert system, Expert Systems with Applications, **158** (2020) 113452.

[37] D. Wang, Y. Yih and M. Ventresca, Improving neighbor-based collaborative filtering by using a hybrid similarity measurement, Expert Systems with Applications, **160** (2020) 113651.