Research Article

Users’ Rating Predictions Using Collaborating Filtering Based on Users and Items Similarity Measures

Sofia Nudrat,1 Hikmat Ullah Khan,1 Saqib Iqbal,2 Mian Muhammad Talha,1 Fawaz Khaled Alarfaj,3 and Naif Almusallam3

1Department of Computer Science, COMSATS University Islamabad, Wah Campus, Wah Cantt 47040, Rawalpindi, Pakistan
2College of Engineering, Al Ain University, Al Ain, UAE
3Department of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia

Correspondence should be addressed to Hikmat Ullah Khan; hikmat.ullah@ciitwah.edu.pk

Received 15 February 2022; Accepted 7 June 2022; Published 8 July 2022

Academic Editor: Thippa Reddy G

Copyright © 2022 Sofia Nudrat et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The social media has made the world a global world and we, in addition to, as part of physical society, are now part of the virtual society as well. There has been the generation of a large amount of information over the social web. By way of increasing online information, new opportunities emerged, and diverse issues have been raised, which have attracted researchers to address these research problems. In this current age, where online business and e-commerce are part of our daily lives, recommender systems (RSs) are very effective for information filtering. RSs play a significant role in our lives by assisting users in recommending items and services what they may be interesting in to purchase or avail. In this research work, our goal is to predict the users’ ratings for various items, which are an active research area in collaborative filtering (CF). In this work, we have explored various similarity measures based on user-user and item-item rating predictions on different datasets by applying collaborative filtering approaches. The comparison of item-item and user-user CF algorithms such as user K-Nearest Neighbour using cosine; similarity, Pearson correlation as well as item-based K-NN using these measures with baseline approaches and matrix-based methods such as Matrix factorization (MF), biased MF, and factor wise MF has been carried out. For empirical-based comparison analysis, diverse approaches have been selected such as slope one, random, and global average, and it revealed that item-item K-NN using Pearson correlation has outperformed all other applied approaches. For the experiments, three real world and widely used datasets of MovieLens 1M, CiaoDVD, and MovieLens 100k have been used. The empirical-based results have been evaluated by using standard performance evaluation measures of RMSE and MAE.

1. Introduction

In current years, online social networking platforms such as Google, Facebook, YouTube, and Twitter have connected people closer to each other. These platforms offer a large amount of online content that makes it very difficult to find desirable or relevant information, and a user may find it difficult to make effective decisions. This problem is commonly referred as the information-overloaded problem. To overcome such a problem, different online social networking platforms provide personalized Recommendation Systems (RSs) [1]. In the present era, RSs are becoming an integral part of many social media platforms or networks and businesses [2]. These RSs give users recommendations on the base of feedback. The feedback in the form of textual reviews or numeric ratings is collected by users using social media sites, after purchasing some items or products. With the help of the nature of reviews, we can understand why customers/users give items with such reviews.

RSs are engines that indicate a particular range of items to a user according to their past experiences, priorities, interest, and preferences [3, 4]. These preferences help to filter the item or content. There are multiple types of RS in daily lives, such as RS for YouTube, Netflix, Amazon, and News aggregators, which help users to have a variety of items on their preferences and help users to save their time and effort. For example, they can get a recommendation for movies, music, food, news, books, and hotels, depending
upon their interest [5]. One of the main issues of an RS is maintaining the user’s interest and recommending the most related content to the user. According to the user’s priorities, if the content has a similarity with his preferences, they can view it; otherwise, they can ignore it.

There are three most popular types of RSs based on the methods used to develop recommender systems such as Collaborative Filtering, Content-Based Filtering, and Hybrid Filtering. Content-Based Filtering (CBF) is among the most adopted filtering techniques in RSs. CBF technique recommends items based on the previously viewed content present in the user’s profile. The main aim of CBF is to suggest the objects or contents that the user likes in the past. CBF performs very efficiently on the recommendation of newly loaded items in the system [5]. CBF prediction technique is a Domain-Dependent approach. It depends upon the metadata of the items. That is, a rich amount of item’s description and a very well-maintained user profile is required for the recommendation task. It emphasizes more on the item’s attribute analysis to develop accurate predictions. A user profile is not that important for CBF because it does not influence the predictions [6].

Collaborative Filtering (CF) makes the prediction quite accurately that the users will agree on their taste in the future as they did in the past [7]. CF Technique is an independent domain technique for prediction. Some content like movies and music that metadata cannot easily describe. Therefore, CF is the best technique to adopt [8]. CF predicts the item by collaborating with the same interest users. For example, users like “Action Movies” to watch and user B also has the same genre. Therefore, user A’s action movies that have been overlooked and rated will be recommended to user B. If user B has not watched them yet. In CF, an Item-User Matrix comprising the user’s preferences are used. This User-Item Matrix database then matches the users with similar interests and choices to make recommendations. This process is also named “Neighborhood” [9]. These recommendations are generated by calculating the resemblance between the user profiles [10].

Hybrid Filtering is the third approach used in developing RS, with different techniques of CBF or CF, or other algorithms when combined from Hybrid Filtering (HF). HF provides optimized solutions to avoid limitations found in CF and CBF pure recommendation systems [11]. The basic idea of HF is that a combination of different approaches can provide better and more accurate predictions in single applied methods. A combined model can reduce the weakness of a single technique [8, 12].

In current years, a huge volume of content is being shared by online users on daily basis. Due to such information explosion, the Internet users may find it difficult to make effective decisions. Thus, to overcome such problems, different social platforms provide personalized recommendations as a basic need for the users. In the personalized recommendation system items are recommended by observing the behaviour and interests of the users through their feedback in the form of ratings, or reviews. By considering users’ ratings, prediction using collaborating filtering based on users and items similarity is carried out in this research.

The rating prediction has been the subject of several research studies. Existing research studies lack to compare item-item and user-user CF algorithms with all mentioned baseline approaches. So, In this research study, our aim is to predict the user-user or item-item rating prediction based on user or item similarity measures. To address this issue, we explore diverse types of RS algorithms related to CF for item-item as well as user-user similarity using diverse formulas such as cosine similarity and Pearson correlation and compare these methods with widely used matrix-based approaches such as MF, biased MF, and factorwise MF. In addition, the baseline approaches such as slope one, random, and global average will be used for comparative analysis. We applied these methods on three widely used datasets and compared them using performance evaluation measures, i.e., RMSE and MAE to compare results.

The main research contribution of this research study includes:

(i) We have explored that item-item-based CF approaches are more effective compared to user-user-based CF approaches for recommendation systems
(ii) It has been figured out empirically that Pearson correlation-based CF approaches show better results compared to cosine similarity-based CF approaches
(iii) The experiments have been performed and these CF based approaches have been compared with matrix factorization-based methods and other standard baseline methods; the Pearson correlation-based CF algorithms outperformed all other algorithms
(iv) The experiments have been performed on three widely used datasets and the evaluation has been carried out using standard performance evaluation measures of RMSE and MAE

The remainder of the article is structured in the following manner. The state-of-the-art approach of RS is presented in Section 2. Section 3 presents the problem statements, and Section 4 presents the methodology; Section 5 presents the experimental setup and performance measurements. The results and discussion are presented in Section 6.

2. Related Work

Let us discuss the previous literature review on recommender systems and RS techniques. To increase the accuracy of recommendation prediction, many approaches for rating predictions are designed to improve the cold start problem and alleviate the sparse data.

2.1. Collaborating Filtering-Based Approaches. In recent years, numerous new CF models have been introduced that compare their accuracy measures with the baseline methods or existing models. The most extensive and commonly applied approach in the recommender systems [13], are CF based methods. In the article, [14], we proposed a collaborative recommendation system with a deep learning method (DLCRS). This method improves the problem of
lack of time sparsity, low ranked approximation, and shortage of meaningful signals. The article makes a comparative analysis between the existing methods and the proposed methods. Results show that proposed approaches give more appropriate results than the existing approaches. In the article, [15], we proposed a novel technique that combines similarity measurement based on ranking and similarity measurement based on structure. The author finds that CFs memory-based combined with similarity measurement shows appropriate accuracy prediction compared to CFs model-based.

The research article [16] proposed a similarity measure multifactor that captures the nonlinear and linear correlation between resulting users from extreme behavior. This method improves the sparse scenario. In the framework of probability matrix factorization and of probability matrix factorization, a fusion approach is given that simultaneously considers global rating information and multifactor similarity. For improving the sparse data optimization, the global rating process with users’ local relations. These are required for extremely sparse data. The result shows that the proposed method improves the sparse data robustness and prediction results compared to the collaborative filtering matrix factorization model. The article [17] proposed a simple linear model named UserReg, based on the matrix factorization (MF) model. This model uses explicit feedback information for rating prediction that helps regularize users’ profile representation, the proposed method applied to the MovieLens dataset, FilmTrust (https://guoguibing.github.io/librec/datasets.html Accessed on May 03, 2021), and Yelp (http://yelp.com/dataset-change/ Accessed on May 03, 2021.). The significant findings of the results showed that UserReg outperforms the baseline methods, as well as it performed comparatively very well compared to other computationally complex models. Moreover, the two robust baseline methods, SVD++ and Biased MF have not been compared with the current novel methods. If both these were compared, the result would favor these two powerful baseline methods.

In the article [18], the author solves the low data utilization in recommendation results and proposed the $\alpha$-divergence based on item similarity measures. This method also influences the rating absolute number and on the computation of results, curated case proportion, which substantially increases the recommendation accuracy. The proposed method is applied to the MovieLens and FilmTrust datasets and shows efficient results. Table 1 shows the comparison of previous work of CF.

### Table 1: Showing the comparison of previous work of CF.

| Year | Method | Dataset | Results |
|------|--------|---------|---------|
| 2020 [14] | Proposed collaborative recommendation system of deep learning method (DLCRS) | MovieLens 1M, Movielens 100k | MovieLens 1M: RMSE 0.903 |
| 2020 [15] | measurement based on ranking and similarity measurement based on the structure | MovieLens 1M, Movielens 100k | RMSE 0.909 MAE 0.708 |
| 2020 [16] | Similarity measure multifactor | CiaoDVD, MovieLens 100k, FilmTrust | RMSE 1.0084 MAE 0.7835 |
| 2021 [17] | Proposed a simple linear model named UserReg, based on the matrix factorization (MF) | MovieLens, FilmTrust, and Yelp | RMSE 0.789 |
| 2021 [18] | Proposed the $\alpha$-divergence based on item similarity measures | MovieLens, FilmTrust | MAE 0.74 RMSE 0.97 |

2.2. Content-Based Approaches. CB recommender system is basically designed to recommend items and objects related to users’ past preferences [19], and only the rated items features and target user ratings have been considered in the approaches of CB [20]. Machine learning different algorithms such as Support Vector Machine (SVM), and KNN has been used in CB recommender systems for implementation purpose. To make recommendations in the CB approach uses information, i.e., social media content. The fields preferred for the approaches of CB are books, movies, and music that can be available as a text [21]. Here, some of last year’s approaches have been discussed.

In article [22], for online shopping, a semantic web mining method for a recommender system has been proposed. The method has two phases. Firstly, textual data preprocessing, the mixture of existing ontology and developed ontology is used. Then, by using the Naïve Bayes algorithm, the recommendation has been generated. The experimental result shows that this method increases 5.2% accuracy. In the article [23], the traditional recommender system context has focused on using Goodreads book and MovieLens 20M datasets. The authors can use a gradient boosting machine (XGBoost) by using content-based and link stream features and present a content-based solely solution. For examination of this model generates a full state-of-the-art recommender system algorithm. The result shows better accuracy. In the present time, online discussions are represented by e-learning systems as a standard for combined learning, which supports the information exchanging and knowledge sharing between learners. The fast development of such mediums has created an issue to find relevant or interesting information. By noticing this problem, the article [24], proposed a new design for e-learning that recommends relevant or interesting information in an online e-learning discussion environment for learners. The design is based on the semantic...
CB recommender system and negative learner’s ratings. The model exploits negative learners’ ratings to achieve a key point in e-learning RSs, which confirms that the items recommended by the recent learners. The experiment was conducted between 25 students. The result shows that this technique gives better results compared to other approaches having similar models.

Group recommender systems have developed a result to recommend useful, suitable, and interesting items to groups of people rather than to individuals. Within this growing environment, such methods have pushed for the introduction of novel recommendation approaches, with the CF paradigm at the foundation of the recommender system. However, CB gives many limitations and drawbacks in these situations such as for several users and item co-occurrence as well as lots of rating values. To overcome this issue, the article [25], we proposed a CB group recommendation system (CB-GRS), and the article analyses and discusses three definite models for the building of CB-GRS: (1) individual ranking and aggregation recommendation, (2) user item matching and aggregation recommendation, and (3) aggregation of the user profile. This research work gives a hybrid CB-GRS that combines methods (2) and (3) and integration function and weighting features. Table 2 shows the existing work of CBF.

The users of social media platforms may influence their thoughts, ideas, and views on a variety of topics while also generating the information that can be used to better understand people’s reactions to certain items. A similar novel research study [26] has been conducted to detect the category of users reviews on social media about restaurants. The proposed method includes next word, next sequence, and pattern prediction algorithms are combined with Aspect Category Detection to create a convolutional attention-based bidirectional modified LSTM. The result of the study favours the proposed technique as it outperforms all the compared state-of-the-art proposed methods. In short, the proposed modified LSTM technique works great for category detection and sentiment analysis as well.

### 2.3. Hybrid-Based Approaches

HB techniques are the combination of different recommendation system techniques to overcome the drawbacks and limitations of such methods [21]. HB method typically combines the CF and CB methods to mix the benefits of such approaches. To achieve higher performance, HB integrates the profiles or features of users, rating of users, and addresses the issue of CF and CB as well [27]. Here, some of last year’s approaches have been discussed. To increase the timeliness and accuracy of the movie, mobile RS, sentiment analysis, and HBRS model are applied on the Apache platform by [28]. In the proposed technique, to obtain the preliminary list, the fusion recommendation method is applied. For enhancing the list, sentiment analysis is applied. In the end, the model is employed on the Apache platform. The results confirm that the proposed method beats the hybrid RS traditional method that integrates CB and CF based on evaluation standards: precision, F-measure, FP rate, and TP rate.

The article [29], proposed a hybrid neural recommendation model to study the representation of items and users deeply from both reviews and ratings. The proposed model contains three main mechanisms such as rating patterns of items and users’ explicit features and learn, a deep rating-based encoder for text review, a model of items and user feedback-based encoder, and a prediction component based on items and users review and rating. In addition, for modelling items and users seeing different in formativeness have different reviews, a new mechanism review-level attention included with representation based on rating as a request vector to choose useful feedback. The result shows that this hybrid model beat the existing techniques.

In the research article [30], the authors proposed a monolithic hybrid recommender system named predictor, that integrates recommender components composed of a fuzzy expert system, CF system, and CB system. The proposed method recommends movies. The method works with users unpopular and favorite genres, using fuzzy expert system movies recommended finally list can be determined which estimates the movie’s importance. The expert method works with numerous constraints such as the number of ratings, level of resemblance between movies rated already, and movie rating average. Therefore, this method achieves better outcomes than existing methods such as the CB system, weighted hybrid system, and CF systems. Table 3 shows the existing work of HB.

Data security is one of the main concerns in today’s world especially if it is related to health records. The lack of security and confidentiality of clinical health records is one of the main issues. A decentralized Personal Health Record (PHR) can be maintained utilizing the Interplanetary File System (IPFS) [31] to provide patients immediate access to their information. However, IPFS has some issues which can lead to permanent data loss. To overcome this issue, a novel blockchain technology based IPFS architecture is proposed that promotes quicker retrieval and consistent PHR.
Several works have been done in rating prediction. The comparison of item-item and user-user CF algorithms such as user K-NN (cosine similarity), user K-NN (Pearson correlation), item K-NN (cosine similarity) and item K-NN (Pearson correlation) with matrix-based methods such as matrix factorization, biased MF and factor wise MF have not done in existing work. No existing work shows the comparison of all the above algorithms.

3. Problem Formulation and Statement

Here, we disuse the problem formulation and problem statement.

3.1. Problem Formulation. The proposed work goal or purpose is to predict unknown item ratings. We apply feature vectors to predict rating. We extract a set of items $I = \{i_1 \ldots i_N\}$ and a set of users $V = \{v_1 \ldots v_M\}$ from datasets. A rating matrix is also set i.e., $R = [R_{v,i}]M \times N$, where $R_{v,i}$ presents item to user rating.

The matrix a gives similarity interest values i.e., $S = [S_{u,v}]M \times M$, where $S_{u,v}$ represents the resemblance interest between user $v$ and user $u$. The matrix a gives similarity values of rating $R = [R_{u,v}]M \times M$, where $R_{u,v}$ represents the rating similarity between users $u$ and $v$. $B = [B_{u,v}]M \times M$ represents the rating interpersonal diffusion behaviour between users. $P = [P_{u,i}]M \times N$ presents the number of personal interests. Our work is to give an item $i \in I$ and user $u \in V$ and $R_{u,i}$ is predicted. By using MF, we calculate the rating prediction of $i$ on $u$.

3.2. Problem Statement. In current years, one of the hottest topics in the big data field is recommender systems, specifically with the growth and development of the World Wide Web. To improve the process of recommendation, several research studies have been carried out. In the marketing and business area, frequently, our decision depends upon the product or product rating or review and other people’s interests or opinions. A higher product rating means more orders and sales. We decide to combine rating prediction and recommendations to predict items that are relevant to the user’s interests.

In our proposed work, we intend to predict item-item or user-user rating prediction by applying collaborating filtering techniques on different datasets. We use similarity-based algorithms that reduce the data sparsity problem and help to predict item-item or user-user rating.

4. Proposed Research Methodology

In this section, the first section gives the discussion about the proposed framework. The second section discussed about the algorithms. Datasets are discussed in the third section and the last section gives the discussion about the performance evaluation measures used to judge the performance.

4.1. Proposed Framework. In the proposed research work, firstly we have downloaded three real-world datasets and have selected the required features for empirical analysis. Then, we have applied state-of-the-art CF algorithms with similarity measures and determine the best one by comparing with the widely used algorithms. Then, we have applied the baseline algorithms on our selected datasets and compared the best algorithm with them. Finally, the evaluation measures are used to evaluate the results. In the end, we have compared the results of existing research work with the proposed method. Figure 1 shows the architecture of our research work.

4.1.1. Features Engineering. Feature engineering shows the features used in the research work for obtaining results. Table 4 shows the symbols and notations used in this document.

4.2. User-User and Item-Item CF Algorithms. Here we explain the user-user and item-item CF algorithms. Followings are the algorithms CF algorithm:

(1) User-to-User Based CF. This technique is used to predict products/items that users like based on the rating given by the other user having similar interests. User-to-user similarity is based on the similarity between two users having the same interest by using distance metrics (Euclidean Distance, Cosine similarity, Pearson’s correlation, etc.) and then find the similar users and recommended products which similar interest the user rated or liked. Similarity $W_{u,v}$ between two users $v$ and $u$, $W_{i,j}$ between two items $j$ and $i$, or is measured by calculating the Pearson correlation using the following equation:

$$w_{u,v} = \frac{\sum_{i \in I} (r_{u,i} - \bar{r}_u)(r_{v,i} - \bar{r}_v)}{\sqrt{\sum_{i \in I} (r_{u,i} - \bar{r}_u)^2} \sqrt{\sum_{i \in I} (r_{v,i} - \bar{r}_v)^2}}$$

(1)

By using (2), we will carry out a prediction for the active user i.e., “a” on definite item $I$ and take the average weighted of all ratings on that item.
We will use the following steps for user-to-user-based similarities and Figure 2 explains these following steps:

1. Load input matrix dataset containing ratings provided by users
2. Find the average rating of all users
3. Then calculate similarity \( w_{u,v} \) between two each two set of users’ \( u \) and \( v \)
4. Predict active user “\( a \)” with definite item \( i \) and take all item rating average
5. Predict rating of the item

\[
P_{a,i} = \frac{r_a + \sum u \in U (r_{u,i} - r_u) W_{a,u} }{\sum_{i \in U} |W_{a,u}|}.
\]  

(2)

By using Equation (4), we will carry out the prediction for the active user, i.e., “\( a \)” on definite item \( i \) and take the average weighted of all ratings on that item.

\[
w_{i,j} = \frac{\sum_{i \in I} (r_{u,i} - r_u)(r_{v,j} - r_v)}{\sqrt{\sum_{i \in I} (r_{u,i} - r_u)^2} \sqrt{\sum_{i \in I} (r_{v,j} - r_v)^2}}.
\]

(3)

We will use the following steps for item-to-item-based similarities and Figure 3 explains these following steps:

1. Take the input of the matrix of ratings of users in the dataset
2. Find the average rating of items
3. Then calculate similarity \( w_{i,j} \) between items \( i \) and \( j \)
4. Predict active user “\( a \)” with definite item \( i \) and take all item rating average
5. Predict rating of the item

(2) Item-to-Item Based CF. Item-to-item similarity is based on the similarity calculation between items. Based on these similarities, products are recommended to those users who have rated or liked similar products in past. The similarity between two users \( i \) and \( j \), \( W_{i,j} \) between two items \( i \) and \( j \), is measured by calculating the Pearson correlation using the following equation:
4.2.1. Modified Cosine Similarity for Rating Prediction.
Cosine similarity is used to calculate the resemblance between items or products. The similarity is calculated between vector angles. By using equation (5), similarities are calculated.

\[ \text{sim}(\overrightarrow{u}, \overrightarrow{v}) = \frac{\overrightarrow{u} \cdot \overrightarrow{v}}{|\overrightarrow{u}| \cdot |\overrightarrow{v}|} \]  

(5)

\( U \): user’s set who have rated items \( i \) and \( j \).

By using Equation (6), the cosine similarity between items is calculated.

\[ \text{sim}(\overrightarrow{i}, \overrightarrow{j}) = \frac{\sum_{u \in U}(r_{ui} - \bar{r}_u)(r_{uj} - \bar{r}_u)}{\sqrt{\sum_{u \in U}(r_{ui} - \bar{r}_u)^2} \sqrt{\sum_{u \in U}(r_{uj} - \bar{r}_u)^2}} \]  

(6)

4.2.2. Baseline Methods. Here, we explain the baseline method used for comparison. Following are the baseline methods.

**Slope One.** Slope one method is a recommended method based on ratings which works on the deviation of the principle intuition between user and items. The algorithm used a simple formula that subtracts the two items average only to determine the deviation. Then, the user’s rating is used for some items, the deviation used in these rating predictions, and other items rated by users. \( f(x) = x + b \) is a predictor. \( \bar{r}_{Bi} = r_{Bi} + (r_{Aj} - r_{Ai}) \) \( r_{Bj} \) is the rating prediction, user \( B \) rated item \( j \). The prediction process has two steps i.e., (1) find deviation \( \text{dev}_{jk} \) between item \( k \) and item \( j \). (2) Find unknown rating \( Pu_{ij} \) that means item \( j \) rated by the target user. Both rate items \( k \) and \( j \) are presented by \( I_{jk} \). \( R_j \) is presented by the target user. Equations (7) and (8) are shown by the following formula:

\[ \text{dev}_{jk} = \sum_{u \in I_{jk}} \frac{r_{ij} - r_{ik}}{\text{card}(I_{jk})} \]  

(7)

\[ P(u)_j = \sum_{k \in I_j} \frac{\text{dev}_{jk} + u_k}{\text{card}(R_j)} \]  

(8)

**Random.** A random algorithm is a method in which as a part of the method logic uses the randomness of sources. This method is basically used to decrease the time complexity or running time, space complexity, or memory used in an algorithm standard. Firstly, the algorithm generates a number randomly i.e., \( r \) in a range of specified numbers and based on \( r \)’s values taking decisions.

4.2.3. Matrix-Based Methods. Here, we explain the matrix-based methods. Following are the matrix-based methods.

**Matrix Factorization.** Items and users are mapped to a dimensionality \( f \) of the joint latent factor space in the MF model, and item-user relations are described as inner products in that space. Every item \( i \) relates to a vector \( q_i \in R^f \) and vector \( P_u \in R^f \) relates to user \( u \). The aspects of \( q_i \) quantify the extent to which an item possesses certain factors, negative or positive, for a given item. The features of \( P_u \) for a particular user \( u \), measure the user’s interest levels that score high on the associated variables, which can be positive or negative. Dot product resulting, \( q_i^T P_u \) represents the user \( u \) and item \( i \) interaction and the whole interest of users in item’s features. Equation (9) shows the formula of matrix factorization.

\[ \hat{r}_{ui} = q_i^T P_u. \]  

(9)

**Biased MF.** On the basis of matrix factorization, the method-based MF [32], which is representing features having explicit rating value by adding a dimensional factor higher that makes the presentation factorized explainable. The method can be used for a variety of discrete features, i.e., age, type, genres, rating, and so on. In Equation (5), \( b_u \) represent bias user and for bias item \( b_i \) is used. A present all items in each feature average to do bias adjustment, where \( a_r \) represents the rating average to value type \( c \), \( b_u,c \) presents bias rating that user type \( c \) prefer, \( b_i,c \) represent the bias popularity of \( i \) item on type \( c \), \( C(i) \) present the value type set that belongs to \( i \), and elements count represented by \( |C(i)| \). Equation (10) shows the formula of biased MF.

\[ \text{bias} = \sum_{c \in C(i)} a_r + b_u,c + b_i,c \]  

(10)

4.3. Datasets. We are taking real-world datasets from movies. In this research study, we have incorporated two benchmark datasets, including MovieLens and CiaoDVD datasets. Both of these datasets are properly balanced and contain real-world users’ ratings over different movies. The sizes of the same domain datasets are different and contain variations of sparsity and rating ranges. These datasets are famous for evaluation purposes in CF recommender systems:

(i) The MovieLens Dataset. The MovieLens dataset consists of two variations: MovieLens 100k (https://grouplens.org/datasets/movielens/100k/ Accessed on May 03, 2021.) ratings and MovieLens 1M (https://grouplens.org/datasets/movielens/1m/ Accessed on May 03, 2021.) ratings. In the dataset of 100k ratings, as the names indicate, 100,000
ratings have been collected from 943 users on 1682 different movies. The range of ratings varies from 1 to 5. Approximately 20 movies have been rated by each user. In the dataset of 1M ratings, 6040 users of the MovieLens platform rated anonymously almost 3900 movies. These datasets are the benchmarks used in many kinds of research.

(ii) The CiaoDVD Dataset. The CiaoDVD (http://konect.cc/networks/librec-ciaodvd-movie_ratings/ Accessed on May 03, 2021.) dataset is a newly collected dataset of the movie domain. The dataset contains 17615 users that have rated 16121 items (movies), and the total ratings are about 700k. The sparsity of the dataset is almost 99.62%. Values of ratings in this dataset are like MovieLens dataset $i - e 1$ (worst) to 5 (best).

4.4. Performance Evaluation Measures. It is used to evaluate the research work. Here, the performance evaluation measures will be discussed in detail:

(i) Root Mean Square Error. RMSE is used to calculate the difference between the values observed and the values predicted. By using Equation (11) to calculate RMSE and $q$ is predicted, and $r$ is a known value.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{m} (q - r)^2}{m}}. \quad (11)$$

(ii) Mean Absolute Error. MAE is used to calculate the difference between predicted and true values. By using equation (12) to calculate this error. $m$ is the number of errors, $y$ is prediction, and $x$ is the true value.

$$\text{MAE} = \frac{1}{m} \sum_{i=1}^{m} |y_i - x|. \quad (12)$$

5. Results and Discussion

Let us discuss here the experimental results using three real world datasets, i.e., MovieLens 100k, MovieLens 1M, and CiaoDVD. In this research work, a comparison of user-user and item-item CF algorithms, i.e., user K-NN (cosine similarity), user K-NN (Pearson correlation), item K-NN (cosine similarity), and item K-NN (Pearson correlation) is performed, and the best algorithm is identified.

Later on, the best algorithms are compared with matrix-based methods, i.e., matrix factorization, biased MF, and factor wise MF. In addition, the best algorithm is further compared with baseline methods including slope one, random, global average, and user item baseline. Then, all methods including CF, MF, and baseline are compared and identified the best algorithm based on performance evaluation measures. Finally, we also compare the best algorithm with existing methods.

5.1. Comparison of User-User and Item-Item CF. The user-user and item-item CF algorithms are applied on the selected datasets and the results are computed as presented in Table 5. The datasets are divided into 70% and 30% for training and testing, respectively, and select features User-Id set as a user identification and ratings set as the prediction. To compute RMSE and MAE values for MovieLens 100k, MovieLens 1M, and CiaoDVD datasets, user K-NN (cosine similarity), User K-NN (Pearson correlation), item K-NN (cosine similarity), and item K-NN (Pearson correlation) are applied. According to the computed results, the item K-NN (Pearson correlation) gives efficient RMSE and MAE compared to other CF methods. Figure 4 can also explain the comparison of collected outcomes.

5.2. Comparison of Matrix-Based Methods with Item-Item (Pearson Correlation). The matrix-based methods are applied on the selected datasets and the results are computed as presented in Table 6. The datasets are divided into 70% and 30% for training and testing, respectively, and select features User-Id set as a user identification and Ratings set as the prediction. To compute RMSE and MAE values for MovieLens 100k, MovieLens 1M, and CiaoDVD datasets, Matrix-Based Methods such as matrix factorization, biased MF, and factor wise MF are applied. In addition, compare the matrix-based method outcomes with item K-NN (Pearson correlation). According to the computed results, the item K-NN (Pearson correlation) gives efficient RMSE and MAE compared to matrix-based methods. Figure 5 can also explains the comparison of collected outcomes.

5.3. Comparison of Baseline Methods with Item-Item (Pearson Correlation). The Baseline Methods are applied on the selected datasets and the results are computed as presented in Table 7. The datasets are divided into 70% and 30% for training and testing, respectively, and select features User-Id set as a user identification and Ratings set as the prediction.

| Algorithms                      | ML 100k | ML 1M | Ciao DVD | ML 100k | ML 1M | Ciao DVD |
|---------------------------------|---------|-------|----------|---------|-------|----------|
| User K-NN (cosine similarity)   | 0.947   | 0.899 | 0.975    | 0.746   | 0.706 | 0.737    |
| User K-NN (Pearson correlation) | 0.939   | 0.887 | 0.969    | 0.739   | 0.696 | 0.737    |
| Item K-NN (cosine similarity)   | 0.934   | 0.883 | 0.975    | 0.736   | 0.694 | 0.736    |
| Item K-NN (Pearson correlation) | 0.933   | 0.879 | 0.964    | 0.734   | 0.690 | 0.734    |
To compute RMSE and MAE values for MovieLens 100k, MovieLens 1M, and CiaoDVD datasets, Slope One, Random, Global Average, and User Item Baseline are applied. In addition, compare the Baseline Method outcomes with item K-NN (Pearson correlation). According to the computed results, the item K-NN (Pearson correlation) gives efficient RMSE and MAE compared to Baseline Methods. Figure 6 can also explains the comparison of collected outcomes.

5.4. Comparison of All Methods Applied to the Above-Mentioned Datasets as a Whole. The comparison of user-user and
item-item CF algorithms, matrix-based methods, and baseline method are applied on the selected datasets and the results are computed as presented in Table 8. The datasets are divided into 70% and 30% for training and testing, respectively, and select features User-Id set as a user identification and Ratings set as the prediction. According to the computed results, the item K-NN (Pearson correlation) gives efficient RMSE and MAE compared to the other CF, matrix-based and baseline methods. Figures 7 and 8 also explain the comparison of collected outcomes.

5.5. Comparison of Proposed Method (Item K-NN Pearson Correlation) with Previous Methods. At the end item, K-NN (Pearson correlation) is also compared with previously existing methods, i.e., UTV [33], DLCRS [14], and GELS [34]. UTV is a User Triple Vector. This vector uses different component similarity and predicts recommender results. DLCRS is a deep learning technique of collaborative RS. DLCRS method matches both movies and users in the same dimensionality and obtains the latent factor. This method used a logistic sigmoid function for prediction. GELS is a

### Table 7: Comparison of BL methods with Item-Item (Pearson Correlation) method.

| Algorithms                        | RMSE  | MAE  |
|-----------------------------------|-------|------|
|                                   | ML 100k | ML 1M | Ciao DVD | ML 100k | ML 1M | Ciao DVD |
| Slope one                         | 0.951  | 0.902 | 1.093    | 0.749    | 0.711 | 0.816    |
| Random                            | 2.105  | 2.138 | 2.307    | 1.723    | 1.762 | 1.888    |
| Global average                    | 1.131  | 1.115 | 1.079    | 0.950    | 0.933 | 0.833    |
| User item baseline                | 0.952  | 0.909 | 0.976    | 0.755    | 0.720 | 0.762    |
| Item K-NN (Pearson correlation)   | 0.933  | 0.879 | 0.964    | 0.734    | 0.690 | 0.734    |

### Table 8: Comparison of Applied Algorithms on Datasets as a whole.

| Algorithms                        | RMSE  | MAE  |
|-----------------------------------|-------|------|
|                                   | ML 100k | ML 1M | Ciao DVD | ML 100k | ML 1M | Ciao DVD |
| User K-NN (cosine similarity)     | 0.947  | 0.899 | 0.975    | 0.746    | 0.706 | 0.737    |
| User K-NN (Pearson correlation)   | 0.939  | 0.887 | 0.969    | 0.739    | 0.696 | 0.737    |
| Item K-NN (cosine similarity)     | 0.934  | 0.883 | 0.975    | 0.736    | 0.694 | 0.736    |
| Item K-NN (Pearson similarity)    | **0.933** | **0.879** | **0.964** | **0.734** | **0.690** | **0.734** |
| Slope one                         | 0.951  | 0.902 | 1.093    | 0.749    | 0.711 | 0.816    |
| Random                            | 2.105  | 2.138 | 2.307    | 1.723    | 1.762 | 1.888    |
| Global average                    | 1.131  | 1.115 | 1.079    | 0.950    | 0.933 | 0.833    |
| User item baseline                | 0.952  | 0.909 | 0.976    | 0.755    | 0.720 | 0.762    |
| Matrix factorization              | 0.988  | 0.954 | 1.107    | 0.771    | 0.746 | 0.860    |
| Biased MF                         | 0.992  | 0.964 | 1.032    | 0.770    | 0.752 | 0.797    |
| Factor wise MF                    | 0.989  | 0.940 | 1.622    | 0.765    | 0.731 | 1.215    |
Gravitational Emulation Local Search. This method considers the galaxy as a search space and imagines every problem solution in the search space as an object. The RMSE achieved values of UTV [31], DLCRS [32], and GELS [33] to applying on MovieLens 1M dataset are 1.130, 0.903, and 0.879, respectively.
The item K-NN (Pearson correlation) RMSE value to apply on MovieLens 1M dataset is 0.897, which shows that the item K-NN (Pearson correlation) gives efficient outcomes compared to the previously existing methods. Figure 9 and Table 9 also explain the comparison of collected outcomes.

The results and discussion may be presented separately or in one combined section, and may optionally be divided into headed subsections.

6. Conclusion and Future Work

In this research work, we predict the users’ ratings for various items, which are an active research area in collaborative filtering. In this work, we explore various similarity measures based on user-user and item-item rating predictions on different datasets by applying collaborative filtering approaches. The comparison of item-item and user-user CF algorithms, i.e., user K-NN (cosine similarity), user K-NN (Pearson correlation), and item K-NN (cosine similarity) with the proposed method item K-NN (Pearson correlation) and matric based methods with item-item i.e., matrix factorization, biased MF, and factor wise MF with proposed method comparison are carried out. In addition, we do the comparison of baseline methods included with item-item including slope one, random, global average, and user item baseline with the proposed method and comparison of the proposed method with all methods above mentioned. Then, in the end, the proposed method’s performance is compared with existing methods. The result shows that the proposed methods give an effective outcome compared to other methods.

Regarding RS, the potential research, future work can be exploring deep learning based latest algorithms such as Recurrent neural networks (RNNs), Convolutional Neural Network (CNN), and Long short-term memory (LSTM) and their variation for the prediction of user’s rating. In addition, the hybrid approach can also be explored, which may consider combining context-based approaches with collaborative filtering approaches.

Data Availability

The datasets used to pursue the findings of this article are freely available online. The URLs of the datasets have been provided in the footnotes in Section 5.1 of Section 5 named as “Experimental Setup.”

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors extend their appreciation to the Deanship of Scientific Research at Imam Mohammad Ibn Saud Islamic University for funding this work through Research Grant no. RG-21-51-01.

References

[1] S. S. Ahmed and U. Hanif, “News Recommendation algorithm based on deep learning,” CRPASE: Transactions of Electrical, Electronic and Computer Engineering, vol. 06, no. 03, pp. 132–140, 2020.
[2] F. Ricci, L. Rokach, and B. Shapira, “Recommender systems: introduction and challenges,” in Recommender Systems Handbook, pp. 1–34, Springer, Boston, MA, 2015.
[3] M. Karimi, D. Jannach, and M. Jugovac, “News recommender systems - survey and roads ahead,” Information Processing & Management, vol. 54, no. 6, pp. 1203–1227, 2018.
[4] M. Elahi, M. Braunofer, F. Ricci, and M. Tkalcic, “Personality-based active learning for collaborative filtering recommender systems,” in Congress of the Italian Association for Artificial Intelligence, Springer, Cham, 2013.
[5] F. Wang, L. P. Casalino, and D. Khullar, “Deep learning in Medicine-Promise, Progress, and challenges,” JAMA Internal Medicine, vol. 179, no. 3, pp. 293–294, 2019.
[6] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, “Recommender systems survey,” Knowledge-Based Systems, vol. 46, pp. 109–132, 2013.
[7] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Item-based collaborative filtering Recommendation algorithms,” Proceedings of ACM World Wide Web Conference, vol. 1, 2001.
[8] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, “Recommendation systems: Principles, methods and evaluation,” Egyptian Informatics Journal, vol. 16, no. 3, pp. 261–273, 2015.
[9] S. A. Amin, J. Philips, and N. Tabrizi, “Current Trends in collaborative filtering Recommendation systems,” in SERVICES, Springer, Cham, 2019.
[10] M. Jalili, S. Ahmadian, M. Izadi, P. Moradi, and M. Salehi, “Evaluating collaborative FilteringRecommender algorithms: a survey,” IEEE Access, vol. 6, p. 1, 2018.
[11] D. Stern, R. Herbrich, and T. Graepel, “Matchbox: large scale online Bayesian recommendations,” in Proceedings of the 18th International World Wide Web Conference, pp. 111–120, ACM, Madrid, Spain, January 2009.
[12] G. Adomavicius and A. Tuzhilin, “‘Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions”, Knowledge and Data Engineering, IEEE Transactions on, vol. 17, pp. 734–749, 2005.
[13] F. Ricci, L. Rokach, and B. Shapira, “Introduction to recommender systems handbook,” in Recommender Systems Handbook, pp. 1–35, Springer, Boston, MA, 2011.
[14] M. F. Aljunid and M. Dh, “An efficient deep learning approach for collaborative filtering Recommender system,” Procedia Computer Science, vol. 171, pp. 829–836, 2020.
[15] D. Wang, Y. Yih, and M. Ventresca, “Improving neighborhood-based collaborative filtering by using a hybrid similarity measurement,” Expert Systems with Applications, vol. 160, Article ID 113651, 2020.
[16] C. Feng, J. Liang, P. Song, and Z. Wang, “A fusion collaborative filtering method for sparse data in recommender systems,” Information Sciences, vol. 521, pp. 365–379, 2020.
[17] H. Zhang, I. Ganchev, N. S. Nikolov, and M. Stevenson, “UserReg: a simple but Strong model for Rating prediction,” 2021, https://arxiv.2012.07601.
[18] Y. Wang, P. Wang, Z. Liu, and L. Y. Zhang, “A new item similarity based on α-divergence for collaborative filtering in sparse data,” Expert Systems with Applications, vol. 166, Article ID 114074, 2021.
[19] P. Lops, M. De Gemmis, and G. Semeraro, “Content-based Recommender systems: state of the art and Trends,” in
Recommender Systems Handbook, pp. 73–105, Springer, Boston, MA, 2011.
[20] C. C. Aggarwal, Recommender Systems, Springer, vol. 1, 2016.
[21] A. Anandhan, L. Shuib, M. A. Ismail, and G. Muitaba, “Social media recommender systems: review and open research issues,” IEEE Access, vol. 6, pp. 15608–15628, 2018.
[22] I. T. Afolabi, O. S. Makinde, and O. O. Oladipupo, “Semantic web mining for content-based online shopping recommender systems,” International Journal of Intelligent Information Technologies, vol. 15, no. 4, pp. 41–56, 2019.
[23] T. Viard and R. Fournier-Sniehotta, “Augmenting content-based rating prediction with link stream features,” Computer Networks, vol. 150, pp. 127–133, 2019.
[24] N. A. Albatayneh, K. I. Gauth, and F.-F. Chua, “Utilizing learners’ negative ratings in semantic content-based recommender system for e-learning forum,” Journal of Educational Technology & Society, vol. 21, no. 1, pp. 112–125, 2018.
[25] Y. Pérez-Almaguer, R. Yera, A. A. Alzahrani, and L. Martinez, “Content-based group recommender systems: a general taxonomy and further improvements,” Expert Systems with Applications, vol. 184, Article ID 115444, 2021.
[26] M. U. Khan, A. R. Javed, M. Ihsan, and U. Tariq, “A novel category detection of social media reviews in the restaurant industry,” Multimedia Systems, vol. 26, pp. 1–14, 2020.
[27] F. Gedikli, D. Jannach, and M. Ge, “How should I explain? A comparison of different explanation types for recommender systems,” International Journal of Human-Computer Studies, vol. 72, no. 4, pp. 367–382, 2014.
[28] Y. Wang, M. Wang, and W. Xu, “A sentiment-enhanced hybrid recommender system for movie recommendation: a big data analytics framework,” Wireless Communications and Mobile Computing, vol. 2018, Article ID 8263704, 2018.
[29] H. Liu, Y. Wang, Q. Peng et al., "Hybrid neural recommendation with joint deep representation learning of ratings and reviews," Neurocomputing, vol. 374, pp. 77–85, 2020.
[30] B. Walek and V. Fojtik, “A hybrid recommender system for recommending relevant movies using an expert system,” Expert Systems with Applications, vol. 158, Article ID 113452, 2020.
[31] A. Mubashar, K. Asghar, A. R. Javed et al., “Storage and proximity management for centralized personal health records using an ipfs-based optimization algorithm,” Journal of Circuits, Systems, and Computers, vol. 31, no. 01, Article ID 2250010, 2022.
[32] W. Sun, X. Zhang, W. Liang, and Z. He, “High dimensional explicit feature biased matrix factorization recommendation,” in Trends and Applications in Knowledge Discovery and Data Mining, pp. 66–77, Springer, Cham, Lecture Notes in Computer Science, 2015.
[33] S. Forouzandeh, K. Berahmand, and M. Rostami, “Presentation of a recommender system with ensemble learning and graph embedding: a case on MovieLens,” Multimedia Tools and Applications, vol. 80, no. 5, pp. 7805–7832, 2021.
[34] T. Mohammadpour, A. M. Bidgoli, R. Enayatifar, and H. H. S. Javadi, “Efficient clustering in collaborative filtering recommender system: hybrid method based on genetic algorithm and gravitational emulation local search algorithm,” Genomics, vol. 111, no. 6, pp. 1902–1912, 2019.