An Enhanced Collaborative Filtering-based Approach for Recommender Systems

Rouhia M. Sallam
Faculty of Computers and Information Technology
Taiz University
Yemen

Mahmoud Hussein
Faculty of Computers and Information
Menoufia University
Egypt

Hamdy M. Mousa
Faculty of Computers and Information
Menoufia University
Egypt

ABSTRACT
Recommender systems are software applications that provide product recommendations for users based on their purchase history or ratings of items. The product recommendations are likely to be of interest to the users and encompass items such as books, music CDs, movies, restaurants, documents (news articles, medical texts, and Wikipedia articles), and other services. In this paper, we propose a framework for collaborative filtering to enhance recommendation accuracy. The proposed approach summarized in two steps: (1) item-based collaborative filtering and (2) singular-value-decomposition-based collaborative filtering. In item-based collaborative filtering, the similarity between the target item and any other item is calculated. Then, the most similar items are recommended. The Singular Value Decomposition based approach handles the problem of scalability and sparsity posed by collaborative filtering and improves the performance of item-based collaborative filtering. We have tested the proposed approach by A Large-Scale Arabic Book Reviews (LABR) dataset. We used four different datasets to compare our approach with existing work. The proposed approach evaluated using the most common metrics found in the collaborative filtering: the mean absolute error (MAE) and the root mean squared error (RMSE). The proposed approach achieved high performance and obtained minimum errors in terms of RMSE and MAE values.

Keywords
Collaborative filtering (CF), k-Nearest Neighbors (KNN), Item-based collaborative, filtering Matrix Factorization (MF) Singular Value Decomposition (SVD), the mean absolute error (MAE), root mean squared error (RMSE).

1.INTRODUCTION
Recommender systems are one of the important techniques in machine learning and data mining, which is used in search of similarities between items and customer preferences. Recommendation techniques are categorized into three types: collaborative filtering, content-based techniques, and hybrid techniques [1].

Collaborative filtering (CF) is the most successful technique in recommender systems. It recommends items by identifying other users with similar tastes; uses their opinion to recommend items to the active user. CF systems have two main approaches: memory-based and model-based approaches [1]. Memory-based approaches use user rating data to compute the similarity between users or items. They are divided into two categories: user-based and item-based CF [2]. User-based CF identifies similar users to the target user for whom the rating predictions are being computed. In item-based CF the similarities need to be computed between items rather than users. In the model-based approaches, machine learning and data mining methods are used for predictive models [2]. Model-based approaches use the ratings to learn a model in order to improve the performance of CF. Examples of these techniques include dimensionality reduction techniques such as singular value decomposition (SVD), the matrix completion technique, latent semantic methods, and regression and clustering [1] .

In the proposed approach, we have combined the best methods in collaborative filtering. Item-based CF provides better performance. The SVD-based approach handles the problem of scalability and sparsity in CF and improves the performance of recommender systems.

In this paper, we analyze the user-item matrix to identify relationships between different items of item-based CF. Then we use these relationships to indirectly compute recommendations for users [3].The item-based approach has two key processes: (a) computing the similarity between each pair of items using various similarity measures like cosine and Pearson metrics [2] and (b) computing the prediction. Item-based CF provides better performance and quality than user-based algorithms in most published research [3, 4,5].

This work uses the model-based technique by applying the matrix factorization algorithm via SVD. Matrix-factorization-based CF aims to reduce the dimensions of the rating matrix and discover potential features under the rating matrix for recommendation [6, 7].

The SVD technique produces high-quality recommendations that handle the problem of scalability and sparsity posed by CF successfully [8, 9, 10, 11, 12].

The experimental results in the proposed approach showed that SVD-based CF achieved better recommendations compared to item-based CF; the two achieved 1.0187 and 1.1969 in terms of RMSE, respectively. They also achieved 0.8077 and 0.922 in terms of MAE, respectively. Our proposed approach also has more accurate when compared to existing work with different datasets.

This paper is organized as follows. Section 2 briefly describes related works in the area of CF. The proposed approach is

described in Section 3. Sections 4 and 5 outline the experimental setting and the results, respectively. Conclusions and suggestions for future work are presented in Section 6.

2.RELATED WORK

In this section, we will describe CF with two methods: memory-based and model-based CF.

Jianfang and Pengfei in [13] introduced a CF algorithm combined with the Singular Value Decomposition (SVD) and Trust Factors (CFSVD-TF). For similarity computation, they used the cosine distance metric. The dataset used was the MovieLens 100k dataset containing 100,000 ratings (1-5) from 943 users for 1682 movies with each user having rated at least 20 movies. The proposed technique was evaluated using the root mean square error (RMSE). The proposed method obtained better prediction accuracy. It obtained 0.9762 in term of RMSE with 10 neighbors.

Another method [14] applied CF based on items to produce a recommendation in movies. The dataset was the Group Lens M1, consisting of around one million ratings from 6,040 users for 4,000 movies. For calculating the similarities between movies, they used adjusted cosine similarity. The proposed approach evaluated using MAE and achieved 0.938 in terms of MAE with 20 neighbors.

The proposed approach in [15] introduced a book recommendation system using item-based collaborative filtering. Cosine distance metrics have been used to calculate similarity books. The dataset used was goodbooks10k contains ratings of 10,000 popular books and 53424 users. The proposed method performed evaluations using MAE. The experimental results achieved 0.72 in terms of MAE.

The proposed in [16] presented the Book Recommendation Algorithm using Deep Learning. The dataset used was goodbooks10k contains 6 million ratings for 10,000 of the most popular books. The experiment randomly divided the data set into an 80% training set and a 10% validation set and a 10% test set. The proposed technique was evaluated using RMSE. It obtained 1.1426 in terms of RMSE.

Mala et al. [17] proposed a web-based movie recommender system that recommends movies to users based on their profile using the different recommendation algorithms such as K-Nearest Neighbor (KNN), singular value decomposition, Alternating Least Squares (ALS) and Restricted Boltzmann Machines (RBM). Experimental results showed that SVD achieved better recommendations compared to KNN, ALS, and RBM. SVD, KNN, and ALS achieved 0.9002, 0.9375 and 1.069 in terms of RMSE respectively. They also achieved 0.6925, 0.7263 0.9935 in terms of MAE respectively.

Sandeep and Rajesh in [18] proposed a new method called Accelerated Singular Value Decomposition (ASVD). It uses momentum based Gradient Descent Optimization. They used real world datasets (MovieLens100k, Film Trust and Yahoo Movie). The proposed technique evaluated using Absolute Error (MAE) and Root Mean Square Error (RMSE). The experimental results showed that the proposed ASVD outperformed other models of SVD using RSVD and SSVD.

3.PROPOSED APPROACH

The main stages for an enhanced collaborative filtering-based approach for recommender systems are shown in Figure 1. The proposed approach consists of two steps: memory-based and model-based CF.

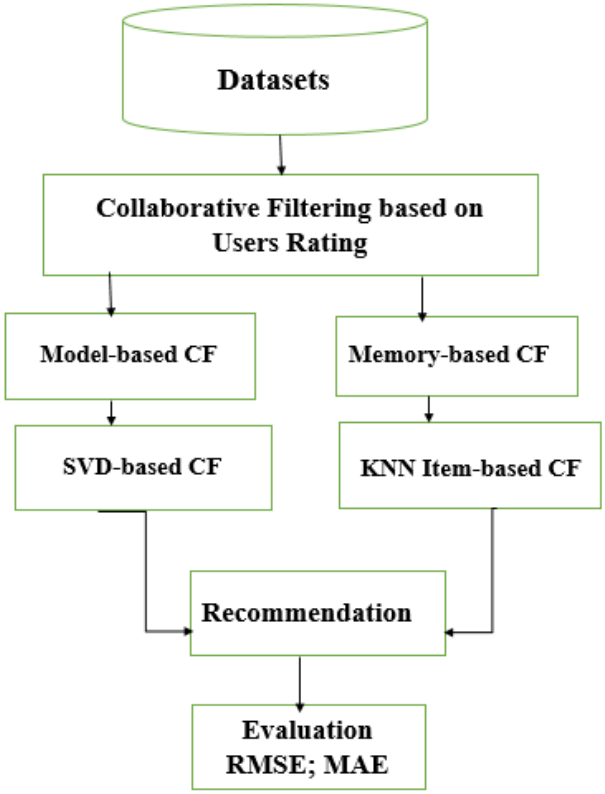

Fig 1: An Enhanced Collaborative Filtering-based Approach for Recommender Systems

3.1 Memory-based Collaborative Filtering

Memory-based methods: are referred to as neighborhood-based CF. The ratings of user-item combinations are predicted based on their neighborhoods. These neighborhoods can be defined in one of two ways: User-based collaborative filtering, and Item-based collaborative filtering. User-based collaborative filtering calculates the similarity between users by comparing their ratings on the same item. Then computes the predicted rating for an item. This method was initially quite popular. They are not easily scalable and sometimes inaccurate [2]. The advantages of memory-based techniques are that they are simple to implement. Other advantages are that the resulting recommendations are often easy to explain. On the other hand, memory-based algorithms do not work very well with sparse rating matrices [2].

The K-nearest neighbors (KNN) Item-based CF finds the similarity between items by selecting the k most similar items. And their corresponding similarities are also determined using a cosine similarity measure. The prediction of the unknown rating is created based on item-item similarity. The top items are returned as recommendations. Figure 2 shows the Collaborative filtering Process.

In the first step, the similarity between two items is measured by computing the cosine of the angle between two vectors m ×n, m list of users and n list of items. The similarity between items i and j, given by [3]. Then the similar neighbor items are

found according to the following similarity.

$$sim(i,j)=cos(\vec{i},\vec{j})=\frac{\vec{i}.\vec{j}}{\|\vec{i}\|2*\|\vec{j}\|2}$$

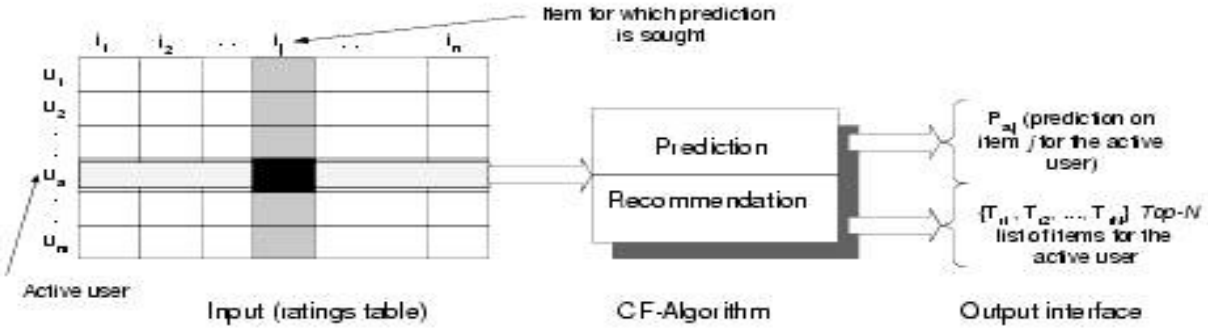

Fig 2: **The Collaborative Filtering Process [3]**

In The next step, the prediction on item *i* for a user *u* is obtained by computing the sum of the ratings given by the user on the items similar to *i*. Each rating is weighted by the corresponding similarity *si*,*j* between items *i* and *j* computed by the following equation [3]:

$$Pu,i=\frac{\sum all\ simila\ r\ items\ ,N(si,N*Ru,N)}{\sum all\ similar\ \ items\ ,N(|si,N|)}$$

Finally, the Top N items are selected using the computed similarly values. These items are not rated by the current user. And recommended to the user.

Despite the success of the Item-Based CF technique, it has some problems such as sparsity and scalability [19]. To solve these problems, we use model-based approach via Matrix Factorization techniques as it deals with these problems successfully and efficiently.

3.2 Model-based Collaborative Filtering

The model-based CF requires a learning phase in advance to learn a model to improve the performance of collaborative filtering. It includes some techniques such as clustering, classification, latent model, Markov decision process (MDP), and matrix factorization.

Matrix Factorization is the most successful latent factor models. It has become popular recently by combining good scalability with predictive accuracy. It maps both users and items to a joint latent factor space of dimensionality f. User-item interactions are modeled as inner products in that space [20]. There are various matrix factorization models: Singular Value Decomposition (SVD), Principal Component Analysis (PCA), Probabilistic Matrix Factorization (PMF) and Non-Negative Matrix Factorization (NMF).

We use SVD as it is one of the most common and successful matrix factorization techniques used in collaborative filtering.

Singular Value Decomposition (SVD): is the powerful technique of dimensionality reduction. This is a specific implementation of the MF approach and is also related to the PCA. The main issue in SVD decomposition is to find a lower-dimensional feature space [20]. SVD of an m × n matrix A is of the form:

$$SVD\ (A) = U\,\Sigma V^T$$

Where, U and V are m × m and n × n orthogonal matrices respectively. Σ is an m × n singular orthogonal matrix with non-negative elements.

An m × m matrix U is called orthogonal if it equals to an m × m identity matrix. The diagonal elements in Σ (σ1, σ2, σ3, … σn) are called the singular values of matrix A. Usually, the singular values are placed in the descending order in Σ. The column vectors of U and V are called the left singular vectors and the right singular vectors respectively. SVD has many desirable properties and is used in many important applications. One of them is the low-rank approximation of matrix A. The truncated SVD of rank k is defined [20, 21]:

$$SVD\ (A_K) = U_K\,\Sigma_K V_K^T$$

Where, U_K ,V_K are m × k and n × k matrices composed by the first k columns of matrix U and the first k columns of matrix V respectively. K × k is the principle diagonal sub-matrix of Σ.. A_K represents the closest linear approximation of the original matrix A with reduced rank k.

4. EXPERMINTAL WORK

In the experimental work, we used the Large-scale Arabic Book Review (LABR) dataset. It has over 63K book reviews, each with a rating of 1 to 5 stars [22].Table 1 is describing the dataset used for testing the proposed approach. Figure 3 shows number of books for each rating.

Table 1: Dataset used in proposed approach evaluation

Number of ratings	63257
Number of unique book id's	2131
Number of unique users	16486
Number of unique reviews	60152
Average number of ratings per user	3.65
Average number of ratings per book	28.23
Average number of reviews per user	3.65

Average number of reviews per book	28.23

Fig 3: Distribution of book ratings

Only three fields were considered to predict user ratings using collaborative filtering: user ID, book ID and rating.

To evaluate the overall performance, we used statistical accuracy metrics that are the most common evaluation measure for prediction accuracy

Statistical accuracy metrics evaluate the accuracy of the system by comparing the numerical recommendation scores with the actual user ratings for the user-item pairs in the test dataset [3].

Mean Absolute Error (MAE) is a metric used to compute the average of all the absolute value differences between the prediction of the algorithm and the real rating [23]. The lower the MAE the better the accuracy. In general, MAE can range from 0 to Infinity, where Infinity is the maximum error depending on the rating scale of the measured application [3,24]. It is computed as follows:

$$MAE=\frac{1}{n}\sum_{i=1}^{n}|pi-qi|$$

Where,

p_i is the actual rating

q_i is the predicted rating

n is the amount of ratings

Root Mean Squared Error (RMSE): computes the mean value of all the differences squared between the true and the predicted ratings. Then, it proceeds to calculate the square root out of the result. RMSE metric is the most valuable metric when significantly large errors are unwanted [23,24]. It is computed as follows:

$$RMAE=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(|pi-qi|)^2}$$

Cross-validation is a validation methodology for analyzing statistical data. Cross-validation splits a dataset into k equally large partitions. One of the partitions is used as test partition while the rest partitions are used as training partitions. The algorithms then train a model with the training partitions and when the training is complete, the model is tested with the test partition, producing test data. This procedure continues until every partition has been the test partition [23].

We divided the datasets into 80% for training and 20% for testing data. Both KNN item-based CF and SVD based CF are evaluated with 5-fold using the LABR dataset. The results are analyzed, interpreted and compared using an absolute mean error and a square root mean error. This will be seen in the results section.

5. RESULTS

This section outlines the experiment results by presenting the obtained MAE and RMSE values using the cross-validation technique. Three experiments are carried out. In the first experiment, we evaluated KNN Item-based CF. In the second one, SVD-based CF is evaluated. In the third experiment, performance comparisons with different methods are performed.

5.1 KNN Item-based CF

In this experiment, the similarity between books is calculated using a cosine similarity metric. We used the LABR dataset and cross-validated the algorithm with 10 as neighborhood size. We run the experiment on the training data and used a test set to compute MAE and RMSE. Table 2 and Figure 4 show the average values in terms of RMSE and MAE are 1.1969, 0.922 respectively.

Table 2: Results of KNN Item-based CF

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
RMSE	1.199	1.190	1.196	1.206	1.193	1.197
MAE	0.921	0.921	0.922	0.932	0.914	0.922

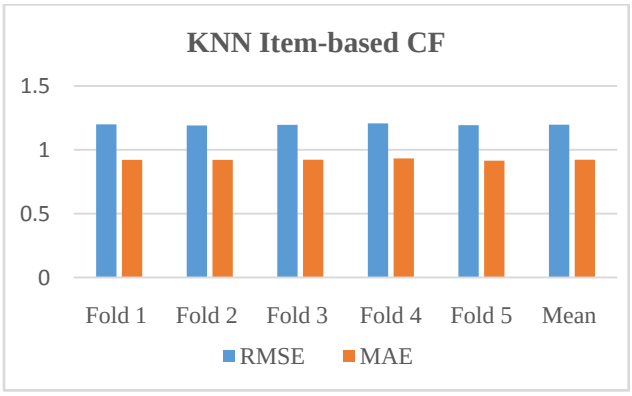

Fig 4: Results of KNN Item-based CF

5.2 SVD-based CF

This experiment presented SVD based CF. The cross-validated over the LABR dataset. We run the experiment on the training data and used a test set to compute RMSE and MAE. Table 3, Figure5 showed the RMSE and MAE scores. It achieved an average of 1.0187, 0.8077 in terms of RMSE and MAE respectively.

Table 3: Results of SVD-based CF

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
RMSE	1.021	1.008	1.020	1.019	1.023	1.019
MAE	0.813	0.801	0.809	0.804	0.810	0.808

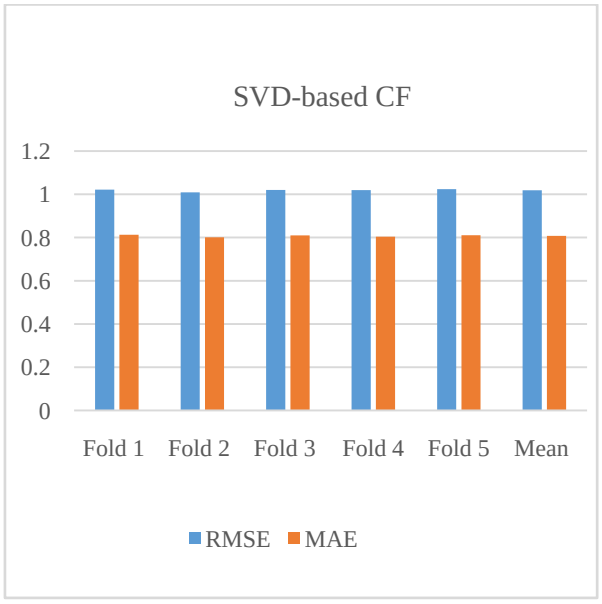

Fig 5: Results of SVD-based CF

We compared the best results achieved by the values of RMSE and MAE. For the memory-based and model collaborative filtering. Figure 6 shows the lowest RMSE and MAE values obtained from SVD based CF comparing by KNN item-based CF that indicates to superior the SVD based CF.

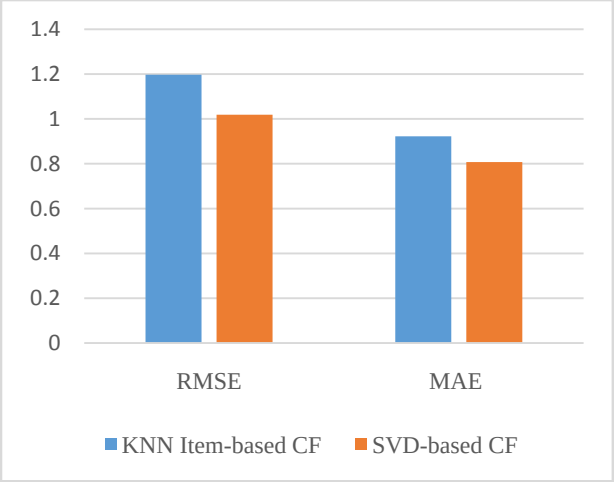

Fig 6: Performance of various CF

5.3 Performance Comparisons with Different Methods

In this section, we present the RMSE and MAE values of each method in the proposed approach and previous works with different datasets. To compare with existing work, we also applied the proposed approach to the data set used in their work.

Table 4: Performance Comparison Results

Method	RMSE	MAE	Datasets
Proposed in [13]	0.9762	-	MovieLens 100k
The proposed approach	0.9365	-	
Proposed in [14]	-	0.938	MovieLens 1M
The proposed approach	-	0.730	
Proposed in [15]	-	0.72	Goodbooks10k
The proposed approach	-	0.67	
Proposed in [16]	1.1426	-	Goodbooks10k
The proposed approach	0.8435	-	
Proposed in [18]	3.432	3.228	MovieLens 100k
The proposed approach	0.9365	0.738	
Proposed in [18]	3.154	2.955	FilmTrust
The proposed approach	0.806	0.625	

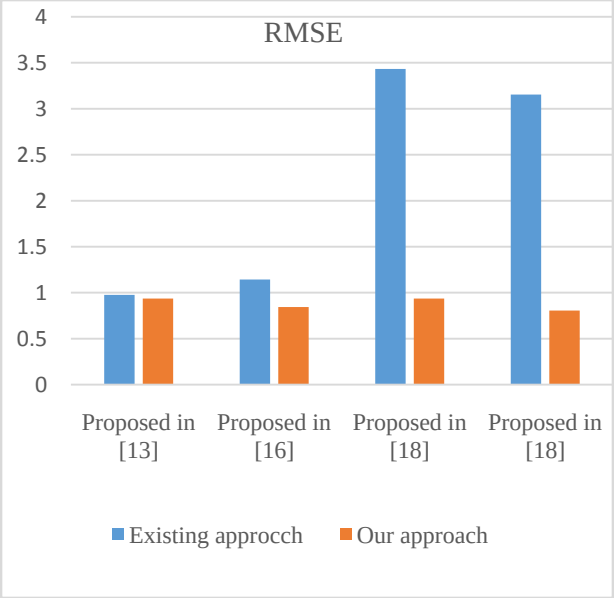

Fig 6: Performance Comparison Results

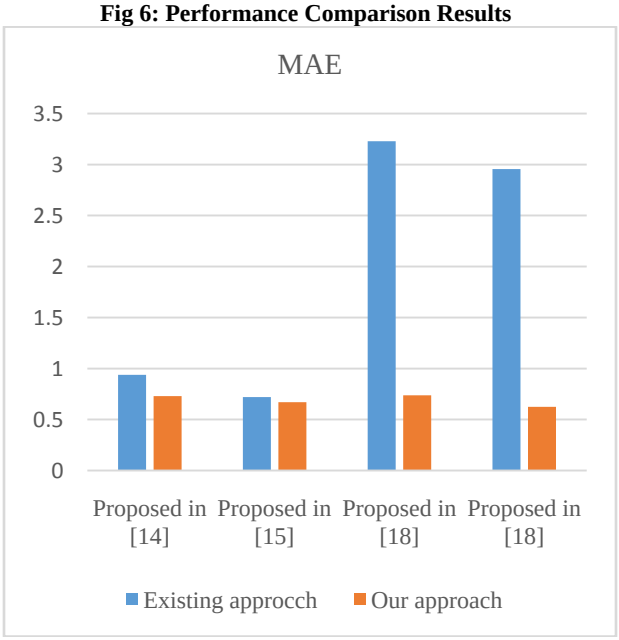

Fig 7: Performance Comparison Results

As presented in Table 4, Figure6 and Figure7 the proposed approach has better performance and accuracy as compared to other methods on different datasets.

6. CONCIUSION

This paper proposed a system for predicting user preferences for items using two types of models: memory-based and model-based collaborative filtering. We used the LABR dataset. Also, we used four different datasets to compare the approach with existing work. The experimental results show that the proposed approach significantly improves the quality and accuracy of the recommendations in terms of RMSE and MAE. The results also showed that SVD-based collaborative filtering is superior to the performance of KNN Item-based CF. The SVD-based CF approach was successful in addressing the problem of scalability and sparsity in KNN Item-based CF. The proposed approach when compared with different methods, it gave the minimum RMSE, MAE values for the rating predictions.

Our future work will focus on experimental evaluating of the proposed collaborative filtering based on sentiment analysis. We will, therefore, evaluate the accuracy and performance of the proposed approach in Arabic datasets.

7. REFERENCES

[1] F. O. Izakaya, Y. O. Foliumin, and B. A. Ojokoh, 2015. "REVIEW Recommendation systems: Principles, methods and evaluation". Egyptian Informatics Journal, 261–273.

[2] Charu C. Aggarwal, 2016, "Recommender Systems", The Textbook, ISBN 978-3-319-29659-3 (eBook), Springer International Publishing Switzerland.

[3] Sarwar B, Karypis G, Konstan J. and Riedl J., 2001, "Item-based collaborative filtering recommendation algorithms Proceedings of the 10th international conference on World Wide Web, 285-295.

[4] Linden. G, Smith. B and York J., 2003, Amazon.com Recommendations: Item-to-Item Collaborative Filtering", IEEE Internet Computing, 76–80 .

[5] P. Prabhu and N. Anbazhagan, 2013, "FI-FCM Algorithm for Business Intelligence", Springer International Publishing Switzerland. 518–528.

[6] Sarwar. B, Karypis.G, Konstan.J and Riedl.J. 2000 "Application of Dimensionality Reduction in Recommender System - A Case Study", in ACM WEBKDD Workshop.

[7] Thi Do. M, Nguyen.D, and Nguyen.L., 2010, "Model-based Approach for Collaborative Filtering", The 6th International Conference on Information Technology for Education, 217-228.

[8] Vozalis. M, Angelos Markos A., and Margaritis K., 2014, "Collaborative Filtering through SVD-Based and Hierarchical Nonlinear PCA", ICANN 2010, Springer-Verlag Berlin Heidelberg, 395–400.

[9] Bokde. D. Sheetal Girase. Sh, Mukhopa. D., 2015, "Matrix Factorization Model in Collaborative Filtering Algorithms: A Survey", Elsevier, the 4th International Conference on Advances in Computing, Communication and Control. Volume 49,136-146.

[10] Hussein M., Okeyo G. and Mwangi.W, 2018, "Matrix Factorization Techniques for Context-Aware Collaborative Filtering Recommender Systems: A Survey", Computer and Information Science; Vol. 11, No. 2; ISSN 1913-8989 E-ISSN 1913-8997.

[11] Bhavanaa P, Kumarb V., and Padmanabhana V., 2019, "Block based Singular Value Decomposition approach to matrix factorization for recommender systems" Pattern Recognition Letters journal homepage (Elsevier).

[12] Yuan X., Han L., Qian S., Guoxia Xu, and Yan H., 2019, "Singular value decomposition-based recommendation using imputed data", Knowledge-Based Systems Volume

163, 485-494.

[13] Wang J., HanP., MiaoY. and Zhang F., 2019, "A Collaborative Filtering Algorithm Based on SVD and Trust Factor", *International Conference on Computer, Network, Communication and Information Systems* .33-39.

[14] Ponnam L. and Punyasamudram S., 2016, "Movie Recommender System Using Item Based Collaborative" Filtering Technique", Proceedings of ICETETS 2016, Kings College of Engineering, 56-60.

[15] Kaivan Shah, 2019, "Book Recommendation System using Item based Collaborative Filtering", *International Research Journal of Engineering and Technology* (IRJET), 5960-5965.

[16] Liu A. S, Gao J., 2019, "Book Recommendation Algorithm Based on Deep Learning", *International Journal of Science*, 152-156.

[17] Saraswat M., Anil Dubey A., Naidu S., Vashisht R. and Singh A., 2020, "Web-Based Movie Recommender System", Springer, Ambient Communications and Computer Systems, 291-301.

[18] Raghuwanshi S., Pateriya R., 2018," Accelerated Singular Value Decomposition (ASVD) using momentum based Gradient Descent Optimization", Journal of King Saud University – Computer and Information Sciences,1-5.

[19] Mohamed M., Khafagy M., Ibrahim M, 2019, "Recommender Systems Challenges and Solutions Survey", International Conference on Innovative Trends in Computer Engineering (ITCE'2019),149-155

[20] Bokde D., Girase S., and Mukhopadhyay D., 2014, "Role of Matrix Factorization Model in Collaborative Filtering Algorithm: A Survey", *International Journal of Advance Foundation and Research in Computer (IJAFRC)* Vol., 1.

[21] Polat H. and Du W. 2005, "SVD-based Collaborative Filtering with Privacy", ACM Symposium on Applied Computing SAC'05, Santa Fe, New Mexico, USA, ACM 1581139640/05/0003, ©ACM, 13-17.

[22] Nabil M., Aly M., Atiya A., 2013, "LABR: A Large-Scale Arabic Book Reviews Dataset", Aclweb. Org. 494–498 (2013).

[23] Najafi S., Salam Z., 2016, "Evaluating Prediction Accuracy for Collaborative Filtering Algorithms in Recommender Systems", KTH Royal Institute of Technology School of Computer Science and Communication.

[24] Herlocker J., Konstan J., Terveen L., and Riedl J., 2004, "Evaluating collaborative filtering recommender systems", ACM Trans. Inf., 5–53.