

# Improving collaborative filtering using lexicon-based sentiment analysis

Rouhia Mohammed Sallam<sup>1</sup>, Mahmoud Hussein<sup>2</sup>, Hamdy M. Mousa<sup>2</sup>

<sup>1</sup>Department of Computer Science, Taiz University, Taiz, Yemen

<sup>2</sup>Department of Computer Science, Menoufia University, Al Minufya, Egypt

## Article Info

### Article history:

Received Jan 3, 2021

Revised Jun 29, 2021

Accepted Jul 29, 2021

### Keywords:

A large-scale Arabic book reviews dataset

Collaborative filtering

LABR lexicon

Sentiment analysis

## ABSTRACT

Since data is available increasingly on the Internet, efforts are needed to develop and improve recommender systems to produce a list of possible favorite items. In this paper, we expand our work to enhance the accuracy of Arabic collaborative filtering by applying sentiment analysis to user reviews, we also addressed major problems of the current work by applying effective techniques to handle the scalability and sparsity problems. The proposed approach consists of two phases: the sentiment analysis and the recommendation phase. The sentiment analysis phase estimates sentiment scores using a special lexicon for the Arabic dataset. The item-based and singular value decomposition-based collaborative filtering are used in the second phase. Overall, our proposed approach improves the experiments' results by reducing average of mean absolute and root mean squared errors using a large Arabic dataset consisting of 63,000 book reviews.

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.*



## Corresponding Author:

Rouhia Mohammed Sallam

Department of Computer Science, Taiz University

Taiz, Yemen

Email: rohya.mohamed@ci.menofia.edu.eg

## 1. INTRODUCTION

Recommender system (RS) handles the overload issues by providing adequate data on a variety of information according to the user preference or the observed actions regarding items [1]–[3]. With the exponential development in e-commerce and social networks, consumers are now contributing by writing reviews, suggestions on some types of products or services, or by putting them online. These reviews make consumers more comfortable as customer opinions are considered to be influential ways for the promotion or demotion of products or services and finding specific information would be easier for customers [4], [5]. The fourth language most widely used on the internet after English, Chinese and Spanish is the Arabic language, according to Internet World Stats on March 31, 2020. It is estimated that by language, there are 237,418,349 Arabic internet users, representing 5.2% of all internet users in the world. Of the estimated 447,572,891 Arabic-speaking people in the world, 53.0% use the internet. In the last 20 years, the number of Internet users who speak Arabic has increased by 9,348.0%, the highest of all other languages. The number of Internet users who speak Arabic has increased by 9,348.0% in the past twenty years, which is the highest of all other languages. Although the Arabic language is considered to be the fourth language, there are existing limited resources for Arabic recommendation systems [6], [7]. This research is a contribution to Arabic recommendation suffering from a lack of research and resources into recommender systems.

We used numeric ratings in collaborative filtering in our previous work [8]. In this work, we expand our work to achieve better recommendation results by using user reviews as ratings after performing the sentiment analysis phase. There are two phases in the proposed approach: sentiment analyses and

recommendations. The first phase uses natural language tools to process user reviews. In order to extract candidate features and personal feelings about each feature, we use a special lexicon large-scale Arabic book reviews (LABR) [9]. After that, extract the score which represents the overall sentiment (positive, or negative) for a review. In the second phase, the recommendation system is used with the sentiment score as a rating. we used the best successful methods of collaborative filtering [8]. The first technique is item-based collaborative filtering (CF) that provides better performance in the memory-based approach [10]–[16]. The second technique is model-based by applying the matrix factorization algorithm via singular value decomposition (SVD). It handles the problem of scalability and sparsity in CF and improves the performance of recommender systems [17]–[20] by using the Item-based and SVD-based CF methods, we have avoided problems found in the previous work, such as scalability and sparsity.

The results of the experimental evaluation of the proposed approach yielded good results as a result of performing the sentiment analysis. The item-based CF method achieved 0.55 and 0.15 in terms of mean absolute error (RMSE) and root mean squared error (MAE), and the SVD-based CF method achieved 0.55 and 0.15 in terms of RMSE and MAE, respectively. Our proposed approach is more accurate compared to relate past work, especially our previous work.

The following is the organization of our paper. The related work is summarized in section 2. Section 3 presents the proposed approach. Sections 4 through 5 provide experimental work and evaluations of results. Finally, the conclusion and future work in section 6.

## 2. RELATED WORK

Recently, recommendation systems have implemented sentiment analysis and opinion mining technologies to define user preferences and optimize recommendation performance. Several researchers have contributed various kinds from these technologies for the English language, but there is very little work on the Arabic recommendation systems. We discuss several of these researches that related to the study of sentiment analysis and collaborative filtering and highlight the works useful for the Arabic language.

Kumar *et al.* [21] suggested a hybrid recommendation system. Collaborative filters have been combined with content-based filters and sentiment analyses. The dataset used consisted of 292,863 Movie Tweetings ratings by 51,081 users in 6,209 different films. The experimental results for average accuracy in the Top 5 and Top 10 recommendations are 0.54, 1.04, 1.86, 3.31, and 2.54, 4.97 respectively for sentiment similarity, hybrid, and the proposed approach. Zhang *et al.* [22] included aspect sentiment collaborative filtering algorithm (ASCF) incorporating the Kano fuzzy model in sentimental analysis. Results of Amazon data set experiments have indicated that ASCF enhances item CF accuracy and collaborative opinion filters well.

Dubey *et al.* have introduced a proposal to enhance the framework of recommendations [23]. They developed a dictionary of feelings to assess the likelihood that feedback will be positive and to determine these feeling values. A collaborative filtering system used sentimental ratings to improve suggestions and filter items with users' generally negative opinions. The datasets used consisted of a MovieLens 100 k dataset and an IMDb review dataset, which consisted of 25,000 positive or negative reviews. The results of the experiments improved compared to the standard recommendation method.

In study [24], a movie recommendation framework was proposed based on the metric of user similarity and opinion mining. In recommending a top-known recommendations list for users, they used aspect-based specific ratings to suggest a top-k recommendation list for users. The dataset used was the 100 k dataset of MovieLens with 100,000 ratings contained 1682 movies, from 943 users. Four metrics for the proposed system have been used: accuracy, recall, precision, and f-measure. The system proposed has achieved greater f-measure efficiency than traditional systems. A recommendation based on the sentimental analysis was introduced by Pradhan *et al.* [25]. They used sentimental analysis to suggest top k reviews based on positive and negative. Experimental results were implemented on Hadoop framework.

The method proposed in [26] presented a recommendation algorithm that enhances collaborative filtration efficiency by quantifying sentiments based on a dictionary. They mixed sentiments with the rating data in order to produce new rating data. The proposed was evaluated by 25,000 reviews. The experimental results in CF use and item-based CF have been improved by 0.059 and 0.0862, respectively while 0.1012, and 0.188 have been improved by the SVD and SVD++ methods in the MAE, respectively. The improvement of 0.0431 and 0.0882 in the RMSE was shown by user-based and item-based CF, while 0.1103 and 0.1756 improved in the methods SVD and SVD++ respectively.

Osman *et al.* introduced a framework to suggest electronic products based on contextual analysis information [27]. The experiment was conducted on a standard available Amazon dataset that containing 2,000 reviews and 5,000 electronic product ratings. The proposed methodology was tested using RMSE and MAE. The proposed model optimizing the traditional sentiment-based model has shown that the

recommendation model has achieved the best RMSE and MAE performance and reduced the level of data sparsity.

Hawashin *et al.* [6] suggested a semantic recommender of Arabic content. The similarity methods used are CHIs, SVDs, and semantic similarity for on Arabic WordNet. The dataset used was a synthesized Arabic dataset and three different stemmers. In terms of MAE with a slightly slower execution time, the results of the experiments showed the CHI-based semantic approach as the best performing approach.

Bader [7] presented a model which recommends users' news by their preferences. He used collaborative and content-based filtering to recommend good Arabic news. He used the stemming operation to Arabic news titles. The dataset used in the system is news collected from various sources of news. He used two ways to test emotion accuracy are electroencephalography (EEG) and SAM methods. The result obtained from the model's EEG is 90%.

Ziani *et al.* [28] have introduced approaches to sentiment analysis with a recommendation system for generating recommendations for users. They used the Semi-supervised support vector machine (S3VM) for opinions polarity score and user-based CF algorithms in the recommendation. The proposed was tested in a variety of datasets: English (2,000) reviews from 50 people, in 40 restaurants (10 users). French data contains five smartphones, 10 users, and 50 evaluations. Arabic with dialect dataset from jumia.com (10 users), 5 Oriental women clothing, and 50 evaluations. The experimental results achieved 0.60 in terms of MAE on Arabic and dialect dataset in terms of MAE in Arabic and dialect datasets, the experimental findings achieved 0.60.

### 3. PROPOSED APPROACH

We expand our previous work [8] for the Arabic book recommendation system that used numeric ratings in CF. By using the sentiment analysis of user reviews, whose purpose is to find the general opinion of the user (positive, negative) for each review. The proposed approach enhances performance quality and reduces effects and dispose of most of the problems in the recommender system and existing works. There are two phases to the proposed approach. The first phase is responsible for analyzing and inferring ratings from user reviews, while the second phase is collaborative filtering that produces item recommendations based on the inferred ratings. Figure 1 provides a summary of the approach proposed. The following subsections provide describing of Arabic sentiment analysis and recommender system in detail.

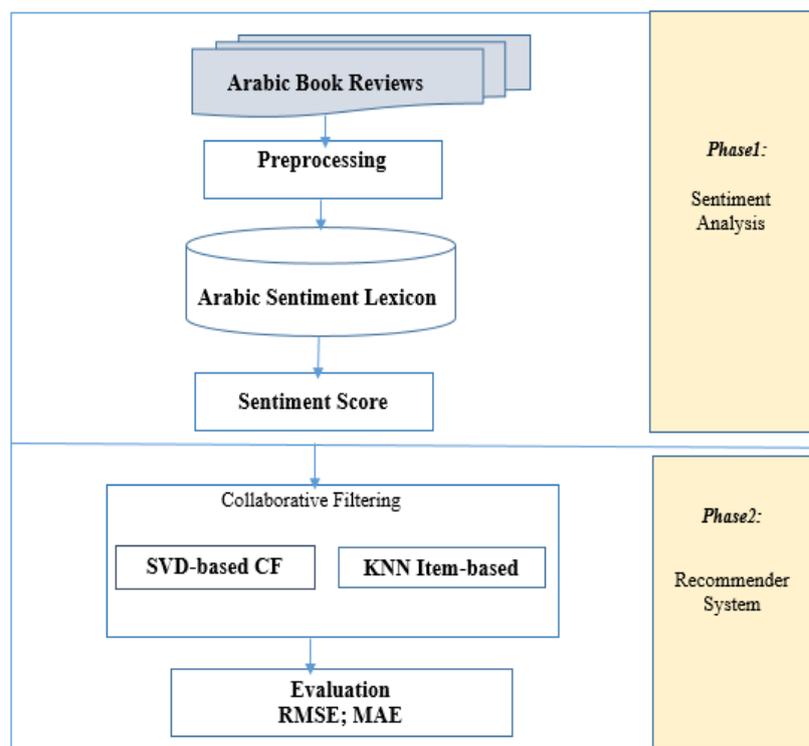


Figure 1. Improving collaborative filtering using lexicon-based sentiment analysis

### 3.1. Sentiment analysis

Sentiment analysis refers to the computational study of the opinions, perceptions, and emotions of people about a product, an entity or an event, or their attributes [29]. Sentiment analysis is the first phase of our approach, which requires many tasks that help identify interesting information in reviews, which are the preprocessing and lexicon-based sentiment analysis.

#### 3.1.1. Preprocessing

Preprocessing is a core aspect of the processing of natural language. Because of the differences in the representation of the text in Arabic. For the preparation of the Arabic text reviews the following steps are used: i) avoid any distortion of characters during the text reading process by converted to UTF-8 encoding; ii) tokenization for text review by segment it into pieces, called tokens, with throwing away some characters at the same time such as punctuation marks, digits, numbers, and special characters; and iii) the stop words in Arabic are removed. Such words (e.g., pronouns and prepositions) are not useful in the Arabic text [30]. For example (... بالإشارة, بالنسبة, الاخرى, ممكن, غيرها, اولئك, بعض, كذلك, كهناك, حاليا, ...).

#### 3.1.2. Lexicon-based sentiment analysis

The lexicon-based approach is an unsupervised method, which relies primarily on lexicons of sentiment. The basic Lexicon approach calculates the number of negative and positive terms in phrases/documents [31]. The text is considered as a positive feeling if several of positive terms are more than the negative [32]. The special LABR dataset manual lexicon is the LABR lexicon [9]. It includes 874 features. The sentiment lexicon LABR contains a word and a corresponding sentiment score to it. Every term from a book review dataset matches the feature of opinion in the special sentiment lexicon for LABR dataset. By aggregating scores of the sentiment in all the terms in the review, the sentiment results are calculated as positive or negative. If the number of positive terms was more than negative, the assessment was considered positive, otherwise negative. It is divided into two classes: positive (+1) or negative (-1) based on the relative number of words in the comment. All of these steps are explained by the algorithm in Figure 2 for determining the overall opinion of any review. Table 1 provides an example of the complete polarity of the book review by taking from the sentiment lexicon and adding the polarity values of each word in the text review.

```

Input: Reviews, LABR lexicon
Output: Sentiment Score
Lexicon=dict ()
Function Senti_Score (reviews)
#Read the Lexicon
Read lexicon (sentiment word)
# Preprocessing reviews
For each review R in Dataset
  Remove stop words
  Score=0
  For each word in review
    If word found in Lexicon
then
      Score=Score + Lexicon
(word)
  End For
  If Score > 0 then
    Score=1
  Else
    Score=-1
  Sentiment score = Sentiment score + Score
  End if
End For
Return: (Sentiment score).

```

Figure 2. Algorithm for calculating sentiment score

### 3.2. Recommender system

In this phase, we applied two collaborative filtering approaches and used sentiment score as the overall rating derived from the method of sentiment analysis. KNN Item-based and SVD based collaborative filtering are these methods.

Table 1. Samples of positive and negative book review

Reviews	Polarity
كتاب جميل أسلوب الطرح رائع Beautiful book and wonderful subtraction style	Positive
قرأت الكتاب بناءً ترشيح شخصين . للأسف لم استمتع بقراءة الرواية وجدتها مملة جداً I read the book based on two people nominated. Unfortunately, I did not enjoy reading the novel and found it very boring.	Negative
من أمتع ما قرأت من روايات بلا شك Of the enjoy novels I have read, without a doubt	Positive
معرفة له محبتوش وعلمي اكتاب اكثر Why I don't like him, I don't know, I felt depressed	Negative

### 3.2.1 KNN item-based CF

The KNN Item-based approach is one of the memory-based CF methods based on searches from close neighbors. KNN Item-based approach discovers the similarities between items. By selecting the most similar items to  $k$ . The similarities are determined by the calculation of cosine similarity. The prediction of the unknown rating is generated based on the similarity between the items. The top items come back as recommendations. In the first, the similarity between two items is measured by computing the cosine of the angle between two vectors  $m \times n$ ,  $m$  list of users, and  $n$  list of items. The similarity between items  $i$  and  $j$ , denoted by  $\text{sim}(i, j)$  is calculated as (1) [10]:

$$\text{Similarity metrics } \text{sim}(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\|_2 \cdot \|\vec{j}\|_2} \quad (1)$$

The next step is the prediction of item  $i$  for a user  $u$  by calculating the total of the user's ratings of things comparable to  $i$ . The equivalent similitudes between items  $i$  and  $j$ , as calculated by (2) [10]:

$$\text{Prediction function: } Pu, i = \frac{\sum \text{all similar items, } N(si, N * Ru, N)}{\sum \text{all similar items, } N(is, N)} \quad (2)$$

Lastly, the Top  $N$  items are selected using similarly computed values not rated by the current user and recommended to the user.

### 3.2.2. SVD-based CF

SVD is one of the most common and successful matrix factorization algorithms used for collaborative filtration. SVD is a powerful reduction technique [33]. The form of the SVD of the  $m \times n$  matrix  $A$  is (3):

$$SVD(A) = U \Sigma V^T \quad (3)$$

An orthogonal  $m \times m$  the same as the matrix with  $m \times m$  is referred to  $U$  matrix. Singular values of matrix  $A$  are known as the Diagonal elements in  $(\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_n)$ . In general, singular values input downward order. The column  $U$  and  $V$  vectors are respectively called the single vectors on the left and the individual vectors on the right. SVD has many attractive characteristics and is used in many key uses. One of them is the low-rank approximation of matrix  $A$ . The truncated SVD of rank  $k$  is defined [34]:

$$SVD(A_k) = U_K \Sigma_K V_K^T \quad (4)$$

where,  $U_K, V_K$  is  $m \times k$  and  $n \times k$  consisting of the first  $k$  columns in  $U$  matrix and the first  $k$  in  $V$  matrix.  $K \times K$  is the origin diagonal sub-matrix of  $A_k$  represents the nearest linear approximation with reduced rank  $k$  to the original matrix  $A$ .

## 4. EXPERIMENTAL WORK

In experimental work, LABR dataset was used. It has over 63 K book reviews [35]. Table 2 outlines the dataset used to test the approach proposed. Figure 3 shows how many reviews are per rating. Only three fields were considered to predict user ratings using collaborative filtering: user ID, book ID and review as rating after performing sentiment analysis.

We used statistical metrics, the most common predictive accuracy test, to evaluate results. Statistical metrics are evaluated the system accuracy by using the numerical recommendation scores in the test dataset

to compare the current user ratings for user-item pairs [10]. These metrics are mean absolute error and root mean squared error:

Table 2. Dataset used in proposed approach evaluation

Number of ratings	63,257
Number of unique book id's	2,131
Number of unique users	16,486
Number of unique reviews	60,152
Average number of ratings every user	3,650
Average number of ratings every book	28,230
Average number of reviews every user	3,650
Average number of reviews every book	28,230

Mean absolute error (MAE) is statistic metric used to calculate the average difference in all the absolute values between prediction and the actual rating [10], [25].

$$MAE = \frac{1}{n} \sum_{i=1}^n |p_i - q_i| \quad (5)$$

Where,  $p_i$  is the actual value, the projected value is  $q_i$  and the ratings are  $n$ .

Root mean squared error (RMSE) is a metric 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 [36], [37]. It is computed as (6):

$$RMAE = \sqrt{\frac{1}{n} \sum_{i=1}^n (|p_i - q_i|)^2} \quad (6)$$

Cross-validation is a method used for the statistical proof verification. Cross-validation divides a dataset into equal size  $k$  divisions. One score is used as a test score while the other divisions are used as training partitions. The algorithms then develop a model with the training partitions, and the model is evaluated on the test partition when the training is complete and test data is produced. This process takes place until every partition is the test partition [36].

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

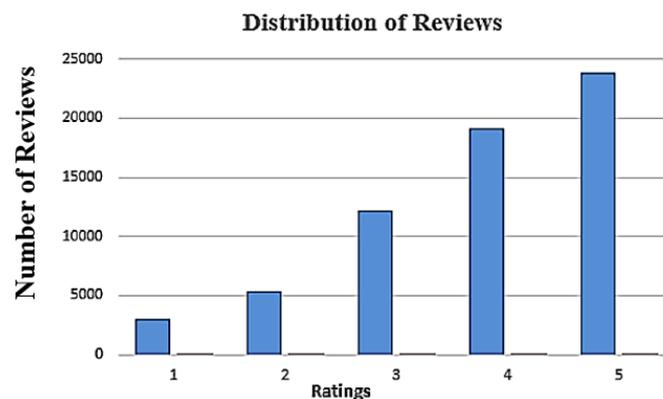


Figure 3. Distribution of reviews

## 5. RESULTS

The research results in this section are summed up by a cross-validation process for presenting data from MAE and RMSE. Three experiments are being conducted. During the first experiment, we evaluated KNN Item-based CF. We evaluated CF on the basis of SVD in the second experiment. The success of the proposed approach was compared with the work already done in the third experiment.

### 5.1. KNN Item-based CF

The similarity between books is measured in this experiment using a cosine similarity metric. We used the LABR dataset. The 10-neighborhood-size algorithm was cross-validated. We ran the experiment with training data and used a test set to calculate MAE and RMSE. The average RMSE and MAE values are shown in Table 3 and Figure 4 are 0.55 and 0.15, respectively.

### 5.2. SVD-based CF

SVD based CF was presented in this experiment. The dataset of LABR was cross-validated. To measure RMSE and MAE, we ran the experiment with training data and used a test set. The RMSE and MAE scores are shown in Table 4, Figure 5. It obtained an average of 0.56 and 0.16 RMSE and MAE, respectively.

Table 3. Results of KNN item-based CF

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
RMSE	0.56	0.55	0.55	0.55	0.56	0.55
MAE	0.15	0.15	0.15	0.15	0.15	0.15

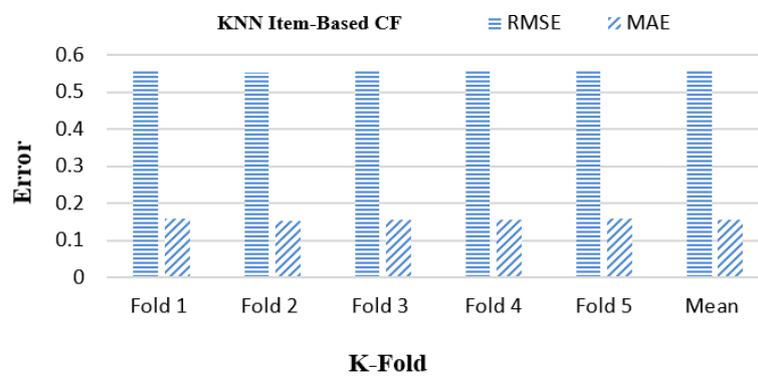


Figure 4. Results of KNN Item-based CF

Table 4. Results of SVD-based CF

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Mean
RMSE	0.53	0.56	0.57	0.56	0.56	0.56
MAE	0.15	0.16	0.17	0.16	0.16	0.16

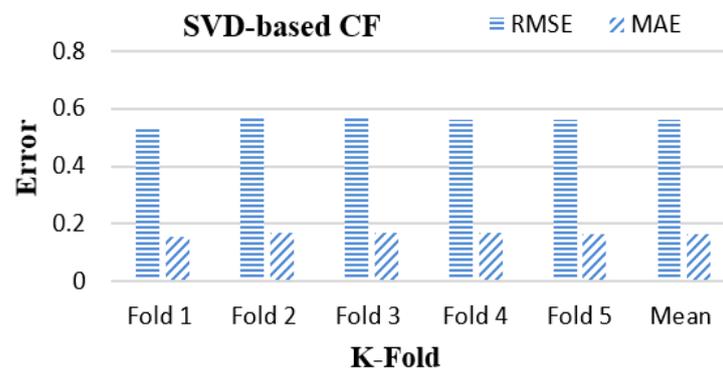


Figure 5. Results of SVD-based CF

### 5.3. Performance comparisons with existing work

The results of our proposed approach with existing work and our previous work are compared in this section using the LABR dataset. Our previous work in [8] for the Arabic book recommendation system used numeric ratings in CF. Ziani's proposed approach in [28] used the user-based CF method to generate

recommendations and used spearman to calculate the similarities between users. In our proposed approach, we used item-based CF, which is more effective and performance than the user-based CF in approach [28]. Also, we used SVD-based CF in the approach proposed, it offers high-quality recommendations and handles the problems of CF scalability and sparsity. In addition, and we used sentiment analysis of user reviews that improved the accuracy of results compare to our previous work in [8]. Table 5 and Figure 6 demonstrate the comparison of the three methods. As presented above our proposed approach has better performance and accuracy as compared to other approaches.

Table 5. Performance comparison results

	Sentiment analysis	Similarity	Methods	RMSE	MAE
Our previous work	-----	Cosine	Item-based CF	1.19	0.92
ZIANI's Approach	S3VM	Spearman	SVD-based CF	1.02	0.80
Proposed Approach	Lexicon-based SA	Cosine	User-based CF	1.0	1.0
			SVD-based CF	0.55	0.15
			SVD-based CF	0.56	0.16

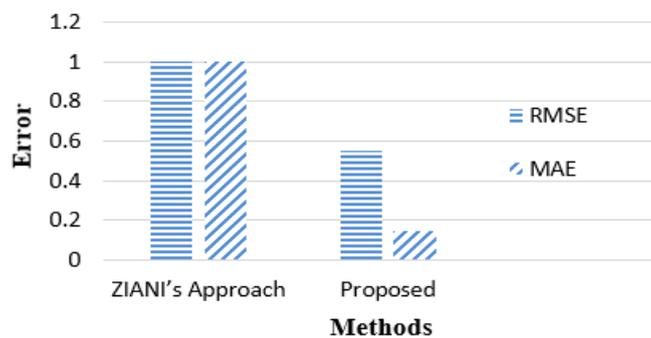


Figure 6. Performance comparison result

## 6. CONCLUSION

In recent years, much work has been dedicated to improving recommendation systems because of the appearance of opinion mining and sentiment analysis techniques. This paper proposed collaborative filtering based on sentiment analysis using an Arabic dataset to provide recommendations for books. The proposed approach improved the accuracy of the Arabic recommendation system and reduced the average error values in terms of RMSE and MAE to 0.5583 and 0.1558, respectively. Compared to previous works on a large Arabic dataset of 63,000 book reviews, our proposed approach yielded better results due to avoiding problems in previous work, such as scalability and sparsity, using KNN component-based CFD and SVD methods. In future work, we will study deep learning in recommendation systems using an Arabic dataset, and attempt to further enhance the performance of our proposed. We will also attempt to solve the issue of cold start.

## REFERENCES

- [1] R. Dridi, S. Zammali, T. Alsulimani, and K. Arour, "Effective rating prediction based on selective contextual information," *Information Sciences*, vol. 510, pp. 218–242, Feb. 2020, doi: 10.1016/j.ins.2019.09.008.
- [2] C. C. Aggarwal, *Evaluating recommender systems*. Recommender Systems, Cham: Springer International Publishing, 2016.
- [3] 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, Nov. 2015, doi: 10.1016/j.eij.2015.06.005.
- [4] L. Chen, G. Chen, and F. Wang, "Recommender systems based on user reviews: the state of the art," *User Modeling and User-Adapted Interaction*, vol. 25, no. 2, pp. 99–154, Jun. 2015, doi: 10.1007/s11257-015-9155-5.
- [5] S. M. Al-Ghuribi and S. A. Mohd Noah, "Multi-criteria review-based recommender system—the state of the art," *IEEE Access*, vol. 7, pp. 169446–169468, 2019, doi: 10.1109/ACCESS.2019.2954861.
- [6] B. Hawashin, S. Alzubi, T. Kanan, and A. Mansour, "An efficient semantic recommender method for Arabic text," *The Electronic Library*, vol. 37, no. 2, pp. 263–280, Apr. 2019, doi: 10.1108/EL-12-2018-0245.
- [7] R. S. Bader, "EANRS: An emotional Arabic news recommender system," in *2019 4th Scientific International Conference Najaf (SICN)*, Apr. 2019, pp. 139–144, doi: 10.1109/SICN47020.2019.9019374.
- [8] R. M. Sallam, M. Hussein, and H. M. Mousa, "An enhanced collaborative filtering-based approach for recommender systems," *International Journal of Computer Applications*, vol. 176, no. 41, pp. 9–15, Jul. 2020, doi: 10.5120/ijca2020920531.
- [9] H. ElSahar and S. R. El-Beltagy, "Building large Arabic multi-domain resources for sentiment analysis," in *International Conference on Intelligent Text Processing and Computational Linguistics*, 2015, pp. 23–34, doi: 10.1007/978-3-319-18117-2\_2.

- [10] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the tenth international conference on World Wide Web - WWW '01*, 2001, pp. 285–295, doi: 10.1145/371920.372071.
- [11] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, Jan. 2003, doi: 10.1109/MIC.2003.1167344.
- [12] P. Prabhu and N. Anbazhagan, "FI-FCM algorithm for business intelligence," Springer International Publishing Switzerland, 2013, pp. 518–528.
- [13] L. Candillier, F. Meyer, and M. Boullé, "Comparing state-of-the-art collaborative filtering systems," in *Machine Learning and Data Mining in Pattern Recognition*, Berlin: Springer Berlin Heidelberg, pp. 548–562.
- [14] K. Verstrepen and B. Goethals, "Unifying nearest neighbors collaborative filtering," in *Proceedings of the 8th ACM Conference on Recommender systems - RecSys '14*, 2014, pp. 177–184, doi: 10.1145/2645710.2645731.
- [15] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, "Improving recommendation lists through topic diversification," in *Proc. of the 14th int. conf. on World Wide Web - WWW '05*, 2005, pp. 22–32, doi: 10.1145/1060745.1060754.
- [16] M. G. Vozalis, A. Markos, and K. G. Margaritis, "Collaborative filtering through SVD-based and hierarchical nonlinear PCA," in *International Conference on Artificial Neural Networks*, 2010, pp. 395–400, doi: 10.1007/978-3-642-15819-3\_53.
- [17] D. Bokde, S. Girase, and D. Mukhopadhyay, "Matrix factorization model in collaborative filtering algorithms: A survey," *Procedia Computer Science*, vol. 49, pp. 136–146, 2015, doi: 10.1016/j.procs.2015.04.237.
- [18] M. H. Abdi, G. O. Okeyo, and R. W. Mwangi, "Matrix factorization techniques for context-aware collaborative filtering recommender systems: a survey," *Computer and Information Science*, vol. 11, no. 2, pp. 1–10, 2018, doi: 10.5539/cis.v11n2p1.
- [19] P. Bhavana, V. Kumar, and V. Padmanabhan, "Block based singular value decomposition approach to matrix factorization for recommender systems," *arXiv preprint arXiv:1907.07410*, Jul. 2019.
- [20] X. Yuan, L. Han, S. Qian, G. Xu, and H. Yan, "Singular value decomposition based recommendation using imputed data," *Knowledge-Based Systems*, vol. 163, pp. 485–494, Jan. 2019, doi: 10.1016/j.knosys.2018.09.011.
- [21] S. Kumar, K. De, and P. P. Roy, "Movie recommendation system using sentiment analysis from microblogging data," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 4, pp. 915–923, Aug. 2020, doi: 10.1109/TCSS.2020.2993585.
- [22] J. Zhang, D. Chen, and M. Lu, "Combining sentiment analysis with a fuzzy kano model for product aspect preference recommendation," *IEEE Access*, vol. 6, pp. 59163–59172, 2018, doi: 10.1109/ACCESS.2018.2875026.
- [23] A. Dubey, A. Gupta, N. Raturi, and P. Saxena, "Item-based collaborative filtering using sentiment analysis of user reviews," in *Int. Conf. on Application of Computing and Communication Technologies*, 2018, pp. 77–87, doi: 10.1007/978-981-13-2035-4\_8.
- [24] N. R. and A. Pethalakshmi, "A novel scheme for movie recommendation system using user similarity and opinion mining," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 9, no. 4S2, pp. 316–322, 2019.
- [25] R. Pradhan, V. Khandelwal, A. Chaturvedi, and D. K. Sharma, "Recommendation system using Lexicon based sentimental analysis with collaborative filtering," in *2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*, Feb. 2020, pp. 129–132, doi: 10.1109/PARC49193.2020.236571.
- [26] T.-Y. Kim, S. B. Pan, and S.-H. Kim, "Sentiment digitization modeling for recommendation system," *Sustainability*, vol. 12, no. 12, Jun. 2020, Art. no. 5191, doi: 10.3390/su12125191.
- [27] N. A. Osman, "Contextual sentiment based recommender system to provide recommendation in the electronic products domain," *Int. J. of Machine Learning and Computing*, vol. 9, no. 4, pp. 425–431, Aug. 2019, doi: 10.18178/ijmlc.2019.9.4.821.
- [28] A. Ziani et al., "Recommender system through sentiment analysis," *2017 2nd International Conference on Automatic Control, Telecommunications and Signals (ICATS)*, 2017.
- [29] F. S. Sathar, "An inheritance-based lexical approach to sentiment analysis," University of Brighton, 2018.
- [30] B. Al-Shargabi, W. Al-Romimah, and F. Olayah, "A comparative study for Arabic text classification algorithms based on stop words elimination," in *Proceedings of the 2011 International Conference on Intelligent Semantic Web-Services and Applications - ISWSA '11*, 2011, pp. 1–5, doi: 10.1145/1980822.1980833.
- [31] C. K. Chung and J. W. Pennebaker, *Linguistic inquiry and word count (LIWC)*. Mahwah: Erlbaum, 2013.
- [32] P. D. Turney and M. L. Littman, "Measuring praise and criticism: Inference of semantic orientation from association," *ACM Transactions on Information Systems (tois)*, vol. 21, no. 4, pp. 315–346, 2003.
- [33] D. K. Bokde, S. Girase, and D. Mukhopadhyay, "Role of matrix factorization model in collaborative filtering algorithm: a survey," *The 4th Int. Conf. on Advances in Computing, Communication and Control (ICAC3)*, vol. 1, pp. 1–12, 2015.
- [34] H. Polat and W. Du, "SVD-based collaborative filtering with privacy," in *Proceedings of the 2005 ACM symposium on Applied computing - SAC '05*, 2005, Art. no. 791, doi: 10.1145/1066677.1066860.
- [35] M. Aly and A. Atiya, "LABR: a large scale Arabic book reviews dataset," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 2013, pp. 494–498, doi: 10.13140/2.1.3960.5761.
- [36] S. Najafi and Z. Salam, "Evaluating prediction accuracy for collaborative filtering algorithms in recommender systems," KTH Royal Institute of Technology School of Computer Science and Communication, 2016.
- [37] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, Jan. 2004, doi: 10.1145/963770.963772.

## BIOGRAPHIES OF AUTHORS



**Rouhia Mohammed Sallam**    is a Ph.D. student at Menoufia University Faculty of Computers and Information, Egypt. She works as a teaching assistant in Faculty of Computers and Information Technology, Taiz University, Yemen. She received the B.S. Degree in Computer Mathematics from Taiz University in 2007 and an M.S. Degree in Computer Science from Menoufia University, Faculty of Computers and Information in 2016. Her main research interest is in Natural Language Processing, Sentiment Analysis, and Recommender system. She can be contacted at email: rohiya79@yahoo.com.



**Mahmoud Hussein**    received his B.Sc. and M.Sc. in Computer Science from Menoufia University, Faculty of Computers and Information in 2006 and 2009 respectively and received his Ph.D. in Software Engineering from Swinburne University of Technology, Faculty of Information and Communications Technology in 2013. His research interest includes software engineering, data mining, machine learning, data privacy, and Security. He can be contacted at email: mahmoud.hussein@ci.menofia.edu.eg.



**Hamdy M. Mousa**    received the B.S. and M.S. in Electronic Engineering and Automatic control and measurements from Menoufia University, Faculty of Electronic Engineering in 1991 and 2002, respectively and received his Ph.D. in Automatic control and measurements Engineering (Artificial intelligent) from Menoufia University, Faculty of Electronic Engineering in 2007. His research interest includes intelligent systems, natural language processing, privacy, security, embedded systems, GSP applications, intelligent agent, bioinformatics, robotics. He can be contacted at email: hamdimmm@hotmail.com.