*Intelligent and Converged Networks*
2023, 4(4): 296−304

ISSN 2708-6240
DOI: 10.23919/ICN.2023.0024

# Design and analysis of a recommendation system based on collaborative filtering techniques for big data

**Najia Khouibiri, Yousef Farhaoui*, and Ahmad El Allaoui**

**Abstract:** Online search has become very popular, and users can easily search for any movie title; however, to easily search for moving titles, users have to select a title that suits their taste. Otherwise, people will have difficulty choosing the film they want to watch. The process of choosing or searching for a film in a large film database is currently time-consuming and tedious. Users spend extensive time on the internet or on several movie viewing sites without success until they find a film that matches their taste. This happens especially because humans are confused about choosing things and quickly change their minds. Hence, the recommendation system becomes critical. This study aims to reduce user effort and facilitate the movie research task. Further, we used the root mean square error scale to evaluate and compare different models adopted in this paper. These models were employed with the aim of developing a classification model for predicting movies. Thus, we tested and evaluated several cooperative filtering techniques. We used four approaches to implement sparse matrix completion algorithms: *k*-nearest neighbors, matrix factorization, co-clustering, and slope-one.

**Key words:** recommendation system; machine learning; collaborative filtering (CF); decision support system; big data

## 1 Introduction

With the advent of big data and technological developments that marked the end of the 20th century and the beginning of this century, the amount of data to be exploited or analyzed has become very voluminous. Knowing what data to look for and where to find them is usually tedious. One such data searching process includes selecting or searching for an online film from a large film database, which makes users spend long hours on the internet or on many movie viewing sites without success until they find a film that suits their taste. Therefore, film recommendation systems aim to assist film lovers by suggesting which movie to watch without going through the lengthy film selection process from a huge series of movies that extends to thousands and millions, which is time-consuming and confusing[1].

In this study, we present a film recommendation system based on Collaborative Filtering (CF) techniques. To this end, we implemented, tested, and evaluated several machine learning algorithms to develop a predictive film provider rating model. The remainder of this paper is organized as follows. A literature review of movie recommendation systems is provided in Section 2. In Section 3, We present the methodology that is employed, along with a discussion on machine learning models and two evaluation metrics. Section 4 discusses the results obtained in this study. Finally, Section 5 presents the conclusions and future studies[2].

## 2 Research background

### 2.1 Related work

Several studies have been conducted to recommend films.

For example, Ref. [3] suggests a movie recommendation system that predicts the user

• Najia Khouibiri, Yousef Farhaoui, and Ahmad El Allaoui are with STI Laboratory, IDM, T-IDMS, Faculty of Sciences and Techniques Errachidia, Moulay Ismail University, Meknes 5003, Morocco. E-mail: n.khouibiri@edu.umi.ac.ma; y.farhaoui@fste.umi.ac.ma; hmad666@gmail.com.
∗ To whom correspondence should be addressed.
Manuscript received: 2023-04-25; accepted: 2023-06-21

preference for a film based on different parameters using the *K*- means clustering and *k*- nearest neighbor (KNN) algorithms.

A hybrid recommendation system proposed in Ref. [4] is built by combining two techniques, CF and content-based filtering (CB), to provide accurate recommendations for movies. The content filtering part of the system has been adopted to train neural networks representing individual user preferences. Filtering results were combined using Boolean and fuzzy aggregation operators. The data adopted in this model led to highly accurate predictions.

Another study constructs a recommendation system based on cosine similarity using KNN with the support of CF technique simultaneously to eliminate the disadvantages of CB filtering[5]. Some scholars suggested the development of a recommendation system based on multiple algorithms to obtain groupings, such as *K*- means, mini-batch *K*-means, birch, affinity propagation, and other algorithms[6]. Additionally, several approaches have been presented to improve *K*- means so that not every cluster can dramatically augment the variance. For movies, this system is restricted to the use of groups based on type and tags.

Most of the above studies employ CF approaches, such as matrix factorization neighborhood-based algorithms[7]. Other methods can be employed to predict missing viewer evaluations and find the list of movies that the user would like to watch. The main contribution of our study is the testing and evaluation of several strategies, including co-clustering and slope-one methods[8].

## 2.2 Recommendation system

The recommendation system is a valuable tool that provides the user with a list of suggestions and directs them to a group of sources that may be useful and interesting to them, which can be difficult to reach in a short period of time within the big data space. For this purpose, one of the following methods is used: CB, CF, or hybrid approaches[9, 10].

### 2.2.1 Content-based filtering

The CB technique is a domain-dependent algorithm that places greater emphasis on the analysis of elements that contribute to generating predictions. For the CB technique, the recommendation is based on the user's profile using features extracted from the content of items that the user has evaluated in the past[11, 12]. Subsequently, it builds a user interest profile (see Fig. 1).

### 2.2.2 Collaborative filtering

CF is an approach based on the sharing of opinions among users. It follows the principle of " word of mouth" that people always practice to build an opinion on a product or service they do not know. The basic premise of this method is that another user's viewpoint can be used to provide a reasonable forecast of preferences to an active user for an item that they have not yet evaluated. This method assumes that if users have the same preferences for a set of items, they will probably have the same preferences for another set of items that they have not evaluated yet[13, 14]. For example, imagine that Ahmed's neighbors discover that a newly opened restaurant in their neighborhood is a success; he will decide to try it. However, if most of his neighbors consider it a failure, he may decide not to go there. Similarly, CF techniques recommend items to the current user that are appreciated by users with the same tastes (see Fig. 2).

### 2.2.3 Hybrid recommendation system

A hybrid recommendation system combines two or more different referral approaches (CF and CB). The earlier approaches had various drawbacks, such as cold start or data scarcity. These issues are frequently resolved by combining two or more techniques. Moreover, with this hybridization, it is feasible to

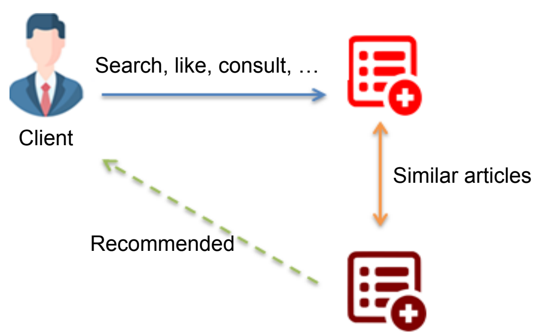



**Fig. 1 Content-based recommendation system.**

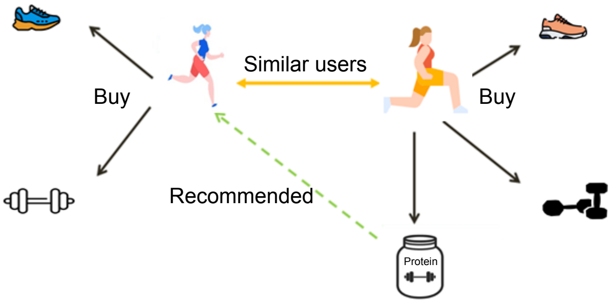



**Fig. 2 Collaborative recommendation system.**

improve the recommendations' quality[15–17].

## 3 Methodology

### 3.1 Data selection

To create recommendations, it is necessary first to search for explicit data that can be worked on. For this purpose and for our choice to be based on a careful study, we rely on the results presented in a previous study[18–20], where the most common and most used dataset was studied by the authors in study recommendation systems[21–23].

As shown in Fig. 3, MovieLens and Amazon datasets are the most popular among researchers and data scientists for conducting diverse experiments, reaching popularities of 40% and 35%, respectively. Because we are concerned with movie recommendations and not with users' opinions about products, it is obvious that our choice will fall in the MovieLens dataset, which is the dataset that we adopt throughout this study[24, 25].

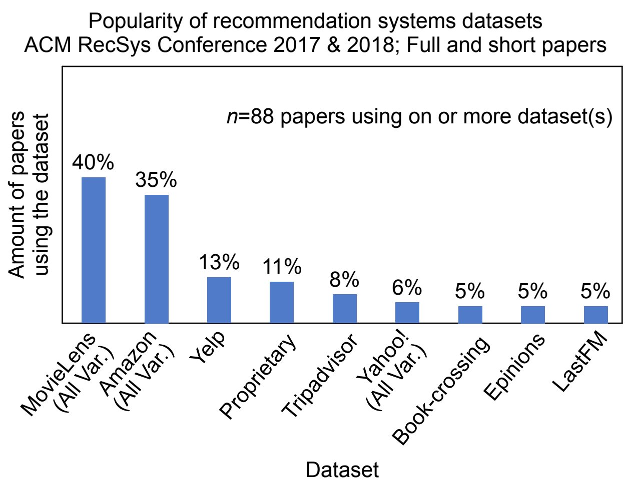



**Fig. 3 Most common datasets for studying recommendation systems.**

### 3.2 Description of the dataset

The MovieLens† dataset contains data from the MovieLens platform collected by the GroupLens Research Laboratory§, a human–computer interaction research laboratory established by the University of Minnesota. Its main purpose includes collecting data to study recommendation systems. The GroupLens Research Laboratory is considered one of the first laboratories to study automated recommendation systems[25, 26]. The MovieLens platform collects user voices, which are then shared through several datasets. The dataset displayed varies in terms of volume and update date[27]. For this study, we visited the GroupLens website and selected one of the recent and recommended data¶ for this research (released on 12/2019), where 25 million ratings and 1 million tag applications were applied to 62 000 films by 162 000 users.

### 3.3 Basic preparation and data exploration

The preparation step is a key initial phase to prepare the dataset for processing and in-depth analysis. Let us import our dataset files first because we will do the same[28].

Now, we have two text files: movies.csv and ratings.csv. We perform a simple exploration of our data to know its content, understand its structure and draw the required statistics for easier interpretation of the results.

As there are no missing values, the data cleaning process will not be performed for this dataset because it is as clean as possible.

Our movies.csv file contains the table of films, which includes three columns: the film identifier (movieId), title, genre, and the title's derived release date. It also contains 62 423 movies.

We do not know the movie selection criteria for the MovieLens dataset, but it seems that most of the films are famous, and others are old, dating back to 1903. Additionally, the sorting of films was based on the date of addition, not the date of release.

The second file, ratings.csv, contains a table of user

† http://www.movielens.org/
§ http://www.grouplens.org/
¶ https://www.grouplens.org/datasets/movielens/25m/

ratings for movies. It has four columns and 697 561 rows. The columns are movieId, rating (users can rate movies from 0.5 to 5), timestamp (the time with the date of voting), and userId. Then, we removed the timestamps column because it serves no purpose for us.

It can be seen that this dataset contains movieId, the title of the film and its genre. We need a dataset containing the userId (to extract user data; thus, we will be able to use user data to increase the precision of recommendations because MovieLens does not offer a table relating to users*)*, movie titles, and notes. This information is included in two different data frame objects: df_ratings and df_movies. To obtain the desired information in a single data frame (Table 1), we can merge these two data frame objects on the movieId column, as it is common for these data frames.

We can do this using the merge() function of the Pandas library.

In our study, finding the best machine learning model that can accurately predict the missing ratings is a difficult task. This is the reason we remove users with only one review (only retain viewers who have more reviews than the average number of reviews per viewer*)*. However, there is nearly 99*%* sparsity in the built movies rating matrix.

### 3.4 Suggested solution

In a study on the health care provider's recommendation system[28], several methods are studied, e.g., the neighborhood method and latent factor models. Additionally, the proposed solution is applied in four stages. In our study, we worked on larger data (including the recommendation system to address the problem of user loss amid big data) and tried to work on some algorithms that are applied in the aforementioned study and test them on a large dataset and other data in a movie recommendation system[29, 30]. The following strategy is suggested for developing a movie recommendation system based on the movie rating model.

**Table 1 Statistic results on the final dataset.**

| Entry | Number |
|---|---|
| Movie | 62 000 |
| Unique user | 100 000 |
| Rating | 400 000 |

(1) Gather data in the form of explicit movie viewer ratings (user ratings) and then prepare and explore it beforehand.

(2) Test and evaluate various machine learning models on ready data using a cross-validation technique and then choose the model with the best performance.

(3) To develop the desired recommendation system, we ultimately deployed the trained model. Figure 4 shows the various phases of the suggested solution.

#### 3.4.1 Machine learning models

Systems using the CF technique should compare objects that are significantly different from one another: items in relation to users. The neighborhood method and latent factor models are the two main strategies for facilitating such a comparative evaluation. Additionally, the co-clustering and slope-one methods have been suggested in the literature to deal with the recommendation issue. The machine learning models that we have employed to forecast missing ratings have been presented in this part.

**Neighbors-based models**. There are two main stages for suggesting recommendations based on the neighbor's model. The first stage is to establish the neighborhood, and the second stage is to make recommendations.

During the neighborhood build process, similarity between users (called a user-based approach) or elements (item-based approach) is measured. The two most widely used similarity measures are Pearson's correlation (PC) coefficient (Eq. (1)) and cosine-based similarity (Eq. (2)).

$$\mathrm{PC}(\boldsymbol{x},\boldsymbol{y})=\frac{\sum_{i=1}^{n}(x_i-x')(y_i-y')}{\sqrt{\sum_{i=1}^{n}(x_i-x')^2}\sqrt{\sum_{i=1}^{n}(y_i-y')^2}} \tag{1}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are two $n$-pointed vectors. The average values of vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ are represented by $x'$ and $y'$, respectively. PC determines the relationship between two sets of data, $\boldsymbol{x}$ and $\boldsymbol{y}$.

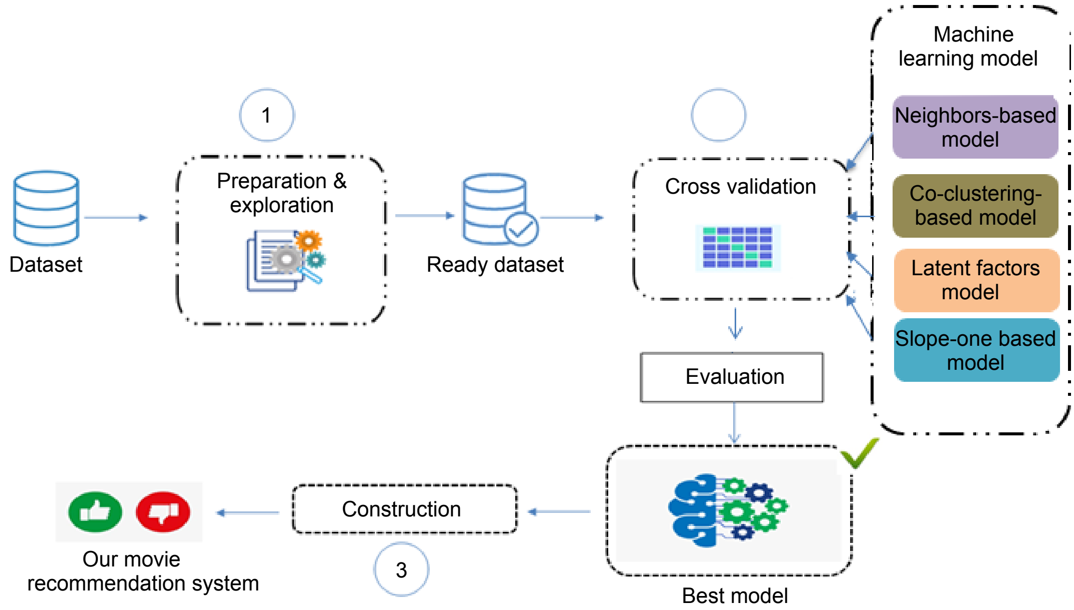



**Fig. 4 Diagram to clarify the suggested solution.**

$$\cos(\theta) = \frac{\sum_{i=1}^{n} E_i \times F_i}{\sqrt{\sum_{i=1}^{n} E_i^2}\sqrt{\sum_{i=1}^{n} F_i^2}} \tag{2}$$

where $E$ and $F$ are two groups of $n$ data points or $n$ characteristic values. $E_i$ and $F_i$ represent the values of feature $i$ in sets $E$ and $F$, respectively. The next phase is to predict user $u$'s evaluation $\hat{r}_{ui}$ that they will most likely give to element $i$. Among the methods that can be used is the use of calculated similarities and corresponding evaluations. Many differences are possible by including biases like means, $Z$-score, or the median user/item evaluations. In this study, we only tested one difference, which is the KNN baseline.

The predicted rating is determined using Eq. (3) if the methodology is user-based; however, if the methodology is item-based, the predicted rating is determined using Eq. (4).

$$\hat{r}_{ui} = \frac{\sum_{v \in N_i^k(u)} \text{sim}(u,v) \cdot r_{vi}}{\sum_{v \in N_i^k(u)} \text{sim}(u,v)} \tag{3}$$

$$\hat{r}_{ui} = \frac{\sum_{j \in N_u^k(i)} \text{sim}(i,j) \cdot r_{uj}}{\sum_{j \in N_u^k(i)} \text{sim}(i,j)} \tag{4}$$

**Latent factor models**. The goal of latent factor models, which are an alternative method to CF, is to identify latent (hidden) characteristics of the data.

This provides explicit explanations about data showing what users are feeling about an element, which is typically stored in ratings matrix form. The matrix factor is among the most popular methods used for identifying latent factors. Figure 5 illustrates the principle of the matrix factorization method. In this study, we limited ourselves to testing only one of the matrix factorization methods, i.e., non-negative matrix factorization (NMF).

By calculating the point product of two vectors related to $\boldsymbol{q}_i$ and $\boldsymbol{p}_u$, It is straightforward to calculate the user's prediction of their evaluation for an element $i$, as indicated in the formula below.

$$\hat{r}_{ui} = \boldsymbol{p}_u^{\mathrm{T}} \boldsymbol{q}_i,$$

where $\boldsymbol{q}_i$ is a vector related to element $i$, and $\boldsymbol{p}_u$ is a vector related to user $u$.

**Slope-one based model**. Slope-one predictors are suggested to use for collaborative rating-based filtering algorithms to reduce the fitting problem, increase efficiency, and facilitate and implement recommendation systems. Based on the use of a simple form of regression, they are regarded as simple approaches to implementing a prediction. The median

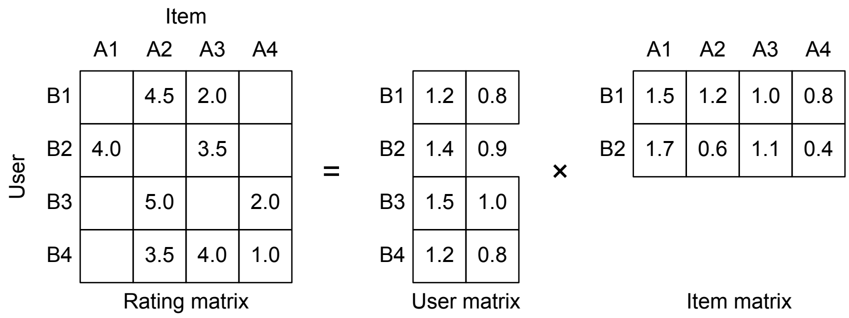



**Fig. 5 Matrix factorization technique.**

contrast between the degrees of the two elements is the only free parameter. In some cases, it has proved to be considerably more precise than the linear regression of the degrees of one element to the degrees of another element.

Therefore, the prediction is calculated using the following relationship:

$$\hat{r}_{u,i}=\mu_u+\frac{1}{|R_i(u)|}\sum_{j\in R_i(u)}\mathrm{dev}(i,j) \tag{5}$$

where $R_i(u)$ is the collection of pertinent elements (i.e., the collection of elements $j$ rated by $u$ and shared with at least one user $i$ ), and $\mathrm{dev}(i,j)$ represents the difference of average rating between elements $i$ and $j$, and it is calculated using the following Eq. (6):

$$\mathrm{dev}(i,j)=\frac{1}{|U_{ij}|}\sum_{u\in U_{ij}} r_{u,i}-r_{u,j} \tag{6}$$

where $U_{ij}$ represents all users that rated items $i$ and $j$.

**Co-clustering-based model**. In the field of data mining, the term "clustering" denotes the process of grouping objects into similar objects belonging to the same group or cluster. Clustering is an unsupervised learning technique. According to the type of data, different aggregation techniques could be applied. The user element rating matrix is used as data in the case of CF. Users and elements are determined by certain $C_{u,i}$ co-clusters, $C_i$ clusters, and certain $C_u$ using a bi-clustering technique. Clusters are selected using an uncomplicated optimization technique, similar to $K$-means[17]. We can calculate predictive rating using the following Eq. (7):

$$\hat{r}_{ui}=\overline{C_{ui}}+\left(\mu_u-\overline{C_u}\right)+\left(\mu_i-\overline{C_i}\right) \tag{7}$$

where $\overline{C_{ui}}$ represents the median evaluation for the $C_{ui}$ co-cluster, $\overline{C_u}$ represents the cluster's median evaluation, to which the user $u$ is considered to belong, and $\overline{C_i}$ represents the average evaluation of the cluster, to which element $i$ is considered to belong.

### 3.4.2 Evaluation metric

The most popular and widely used scales for evaluating recommendation systems are the root mean squared error (RMSE) and mean absolute error scales. In this study, we used the RMSE scale to evaluate our recommendation system, which can be calculated using Eq. (8).

$$\mathrm{RMSE}=\sqrt{\frac{\sum(\hat{r}_{ui}-r_{ui})^2}{n}} \tag{8}$$

where $\hat{r}_{ui}$ is the expected user $u$ rating for item $i$, $r_{ui}$ is the rating that was actually given, and $n$ is the volume of the test set (size).

In this paper, on all of our samples, we perform a 5-cross-validation RMSE. We try to train our model using 80% of the data, and the remaining 20% is used for testing the accuracy.

## 4 Result

The outcomes of our testing and evaluation of various methods of the primary models mentioned above are summarized in Table 2.

It can be seen that the baseline user-based CF KNN is the best model in terms of RMSE value. Our aim is to find the best (optimal) metrics for each of the models. A detailed summary of the findings is provided in Table 2.

## 5 Conclusion and perspective

This study aimed to develop recommendation systems using the CF approach with several machine learning models. Our testing experiments proved that the

**Table 2 Results summary.**

| Metric | Neighbors-based model | Co-clustering based model | Latent factors based model | Slope-one based model |
|---|---|---|---|---|
| Tested approach | User-based CF KNN + baseline | Co-clustering based on *K*-means algorithm | NMF | Basic slope-one algorithm |
| RMSE | 0.8535 | 0.9131 | 0.8746 | 0.8864 |

models based on neighbors and latent factors have succeeded in providing more accurate recommendations (i.e., with a low error rate). Other methods can also be adopted, such as slope-one and co-clustering, to solve the problem of anticipated missing ratings in the rating matrix for films by users.

However, our proposed recommendation system suffers from several obstacles, including the method of calculating similarities and cold start obstacles, which can only be solved by adopting more than one machine learning technique (there is a need to create a hybrid system) or relying on written surveys of people. This is done by merging survey programs with movie recommendation programs.

The task of recommending movies has been daunting, and it will become more challenging in the years to come because of the alarming increase in the volume of data. This means that the basis for recommending movies should not be limited only to the opinions of similar users, but more information, such as age and gender, should be considered. Why not also consider the health status of the user? This may sound somewhat strange, but it is realistic. How many users have died just because they watched a comedy or horror movie recommended to them by a similar friend, without taking into account that they are asthmatic or heart patients? Moreover, how many teenagers, due to the error of recommending films that do not agree with their age, have committed suicide?

In essence, the more information we collect, the greater the significance of similarity calculations, the recommendation system is more accurate and safer for the user’s life because human life does not accept any room for error. Therefore, relying solely on machine learning techniques is insufficient. Instead, we must look forward to developing a hybrid recommendation system that adopts various deep learning techniques and integrates data mining techniques to eliminate the cold start problem.

## References

[1] F. Furtado and A. Singh, Movie recommendation system using machine learning, *Int. J. Res. Ind. Eng*., vol. 9, no. 1, pp. 84–98.

[2] A. Hessane, A. El Youssefi, Y. Farhaoui, B. Aghoutane, N. A. Ali, and A. Malik, Healthcare providers recommender system based on collaborative filtering techniques, in *Machine Learning and Deep Learning in Medical Data Analytics and Healthcare Applications*, O. P. Jena, B. Bhushan, and U. Kose, Eds. Boca Raton, FL, USA: CRC Press, 2022. pp. 261–274.

[3] R. Ahuja, A. Solanki, and A. Nayyar, Movie recommender system using K-means clustering AND K-nearest neighbor, in *Proc. 2019 9th Int. Conf. Cloud Computing, Data Science & Engineering* (*Confluence*), Noida, India, 2019, pp. 263–268.

[4] C. Christakou, S. Vrettos, and A. Stafylopatis, A hybrid movie recommender system based on neural networks, *Int. J. Artif. Intell. Tools*, vol. 16, no. 5, pp. 771–792, 2007.

[5] M. Gupta, A. Thakkar, Aashish, V. Gupta, and D. P. S. Rathore, Movie recommender system using collaborative filtering, in *Proc. 2020 Int. Conf. Electronics and Sustainable Communication Systems* (*ICESC*), Coimbatore, India, 2020, pp. 415–420.

[6] D. Cintia Ganesha Putri, J.-S. Leu, and P. Seda, Design of an unsupervised machine learning-based movie recommender system, *Symmetry*, vol. 12, no. 2, p. 185, 2020.

[7] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, Recommendation systems: Principles, methods and evaluation, *Egypt. Inform. J.*, vol. 16, no. 3, pp. 261–273, 2015.

[8] I. Benouaret, Un système de recommandation contextuel et composite pour la visite personnalisée de sites culturels, (in French), Ph. D. dissertation, University of Technology of Compiègne, France, 2017, pp. 181.

[9] M. Baidada, K. Mansouri, and F. Poirier, Hybrid filtering recommendation system in an educational context, *Int. J. Web Based Learn. Teach. Technol.*, vol. 17, no. 1, pp. 1–17, 2022.

[10] J. Beel and V. Brunel, Data pruning in recommender systems research: Best-practice or malpractice? in *Proc. ACM RecSys 2019 Late-Breaking Results & 13th ACM Conf. Recommender Systems*, Copenhagen, Denmark,

2019, pp. 26–30.

[11] J. Bastin, Etude des systèmes de recommandations et mise en pratique des algorithms, (in French), master dissertation, University of Liège, Belgium, 2020.

[12] Y. Koren, Factorization meets the neighborhood: a multifaceted collaborative filtering model, in *Proc. 14th ACM SIGKDD Int. Conf. Knowledge discovery and data mining*, Las Vegas, NV, USA, 2008, pp. 426–434.

[13] O. A. Montesinos-López, A. Montesinos-López, J. Crossa, J. C. Montesinos-López, D. Mota-Sanchez, F. Estrada-González, J. Gillberg, R. Singh, S. Mondal, and P. Juliana, Prediction of multiple-trait and multiple-environment genomic data using recommender systems, *G3 Bethesda Md*, vol. 8, no. 1, pp. 131–147, 2018.

[14] P. Wang and H. Ye, A personalized recommendation algorithm combining slope one scheme and user based collaborative filtering, in *Proc. 2009 Int. Conf. Industrial and Information Systems*, Haikou, China, 2009, pp. 152–154.

[15] D. Lemire and A. MacLachlan, Slope one predictors for online rating-based collaborative filtering, in *Proc. 2005 SIAM Int. Conf. Data Mining*, Newport Beach, CA, USA, 2005, pp. 471–475.

[16] M. Al-Ghamdi, H. Elazhary, and A. Mojahed, Evaluation of collaborative filtering for recommender systems, *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 3, pp. 559–565, 2021.

[17] T. George and S. Merugu, A scalable collaborative filtering framework based on co-clustering, in *Proc. 5th IEEE Int. Conf. Data Mining* (*ICDM'05*), Houston, TX, USA, 2005.

[18] G. Shani and A. Gunawardana, Evaluating recommendation systems, in *Recommender systems handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA, USA: Springer, 2011, pp. 257–297.

[19] Y. Farhaoui, Design and implementation of an intrusion prevention system, *Int. J. Netw. Secur.*, vol. 19, no. 5, pp. 675–683, 2017.

[20] Y. Farhaoui, S. Ojo, L. A. Akinyemi, and A. L. Imoize, Editorial, *Big Data Mining and Analytics*, vol. 6, no. 3, pp. i–ii, 2023.

[21] Y. Farhaoui, Intrusion prevention system inspired immune systems, *Indones. J. Electr. Eng. Comput. Sci.*, vol. 2, no. 1, p. 168, 2016.

[22] Y. Farhaoui, Big data analytics applied for control systems, in *Proc. Int. Conf. Advanced Information Technology, Services and Systems* (*AIT2S-17*), Tangier, Morocco, 2017, pp. 408–415.

[23] Y. Farhaoui, B. Bhushan, M. Fattah, and B. Aghoutane, Editorial, *Big Data Mining and Analytics*, vol. 5, no. 4, pp. i–ii, 2022.

[24] S. S. Alaoui, Y. Farhaoui, and B. Aksasse, Hate speech detection using text mining and machine learning, *Int. J. Decis. Support. Syst. Technol.*, vol. 14, no. 1, pp. 1–20, 2022.

[25] S. S. Alaoui, Y. Farhaoui, and B. Aksasse, Data openness for efficient e-governance in the age of big data, *Int. J. Cloud Comput.*, vol. 10, nos. 5&6, p. 522, 2021.

[26] A. El Mouatasim and Y. Farhaoui, Nesterov step reduced gradient algorithm for convex programming problems, in *Proc. 3rd Int. Conf. Big Data and Networks Technologies* (*BDNT 2019*), Leuven, Belgium, 2019. pp. 140–148.

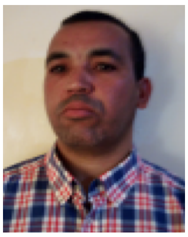
[27] A. Tarik, and Y. Farhaoui, Recommender system for orientation student, in *Proc. 3rd Int. Conf. Big Data and Networks Technologies* (*BDNT 2019*), Leuven, Belgium, 2019, pp. 367–370.

[28] S. Sossi Alaoui, Y. Farhaoui, and B. Aksasse, A comparative study of the four well-known classification algorithms in data mining, in *Proc. Int. Conf. Advanced Information Technology, Services and Systems* (*AIT2S-17*), Tangier, Morocco, 2017. pp. 362–373.

[29] Y. Farhaoui, Teaching computer sciences in Morocco: An overview, *IT Prof.*, vol. 19, no. 4, pp. 12–15, 2017.

[30] Y. Farhaoui, Securing a local area network by IDPS open source, *Procedia Comput. Sci.*, vol. 110, pp. 416–421, 2017.

**Najia Khouibiri** received the MEng degree in decision information systems and imaging from Moulay Ismail University, Morocco in 2022. She is a PhD candidate at the Engineering Science and Technology Laboratory, IDMS Team, Faculty of Sciences and Techniques Errachidia, Moulay Ismail University, Morocco. Her research interests include business intelligence, cloud computing, and big data.



**Ahmad El Allaoui** is now an assistant professor at the Department of Computer Science, Faculty of Sciences and Techniques Errachidia, Moulay Ismail University, Morocco. He is an IDMS Team member. He focuses in semantic image segmentation, medical imaging, classification algorithms, segmentation, image processing, evolutionary algorithms, and genetic algorithms.

**Yousef Farhaoui** obtained the PhD degree in computer security from Ibn Zohr University of Science, Morocco in 2012. He is now a professor at Faculty of Sciences and Techniques Errachidia, Moulay Ismail University, Morocco. He is the chair of IDMS Team and director of STI Laboratory. He is the local publishing and research coordinator, Cambridge International Academics Ltd., UK. His research interests include learning, e-learning, computer security, big data analytics, and business intelligence. He has published 3 books on computer science. He is also a member of the scientific committee of several international congresses and a member of various international associations. He has authored 7 books and many book chapters with reputed publishers, such as Springer and IGI. He is a reviewer for IEEE, IET, Springer, Inderscience, and Elsevier journals. He is also the guest editor of many journals with Wiley, Springer, and Inderscience. He has been the general chair, session chair, and panelist in several conferences. He is a senior member of IEEE, IET, ACM, and EAI Research Group.