

## Employing opposite ratings users in a new approach to collaborative filtering

Abdellah El Fazziki<sup>1</sup>, Yasser El Madani El Alami<sup>2</sup>, Jalil Elhassouni<sup>3</sup>, Ouafae El Aissaoui<sup>1</sup>,  
Mohammed Benbrahim<sup>1</sup>

<sup>1</sup>LIMAS, Faculty of Sciences Morocco, Sidi Mohamed Ben Abdellah University, Fez, Morocco

<sup>2</sup>Department of Computing Science, Mohammed V University Morocco, Rabat, Morocco

<sup>3</sup>Rabat IT Center, Faculty of Sciences, Mohammed V University Morocco, Rabat, Morocco

### Article Info

#### Article history:

Received Mar 31, 2021

Revised Nov 3, 2021

Accepted Nov 27, 2021

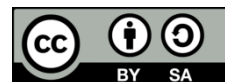
#### Keywords:

Collaborative filtering  
Gray sheep problem  
Imaginary user  
Opposite neighbors  
Recommendation system  
Similarity measure

### ABSTRACT

Over the past few decades, various recommendation system paradigms have been developed for both research and industrial purposes to satisfy the needs and preferences of users when they deal with enormous data. The collaborative filtering (CF) is one of the most popular recommendation techniques, although it is still immature and suffers from some difficulties such as sparsity, gray sheep and scalability impeding recommendation quality. Therefore, we propose a new CF approach to deal with the gray sheep problem in order to improve the predictions accuracy. To realize this goal, our solution aims to infer new users from real ones existing in datasets. This transformation allows for creating users with opposite preferences to the real ones. On the one hand, our approach permits to amplify the number of neighbors, especially in the case of users who have unusual behavior (gray sheep). On the other hand, it facilitates building a dense similar neighborhood. The basic assumption behind this is that if user X is not similar to user Y, then the imaginary user  $\neg X$  is similar to the user Y. The performance of our approach was evaluated using two datasets, MovieLens and FilmTrust. Experimental results have shown that our approach surpasses many traditional recommendation approaches.

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



### Corresponding Author:

Abdellah El Fazziki

LIMAS, Faculty of Sciences Fez, Sidi Mohamed Ben Abdellah University

Fez, Morocco

Email: elf.abdellah@gmail.com

## 1. INTRODUCTION

Recommendation systems are smart tools that can recommend things to users based on their preferences [1], especially personalized recommendation system. The primary purpose of recommendation systems is to facilitate the task of users by providing them with items responding to their needs. The key of personalized recommendation systems is the user preferences. In fact, preferences allow recommendation systems to understand users' needs and behavior. They are mainly based on users' history (ratings and clicks); these interactions are divided into two categories: implicit feedback data (e.g., clicks, purchases) and explicit feedback data (e.g., ratings, votes). Explicit feedback data are more widely used in the research fields of recommendation system [2], [3]. With the intensifying problem of information overload, recommendation systems have become indispensable for users to find accurate information, products, or services they are seeking. These systems have the role of filtering incoming information by transmitting relevant flows to the user and blocking those that are irrelevant [4].

In the last decades, in the last decades, online industries have intensively used the recommendations systems to claim their place in the market and improve their customer relationship management. For instance, e-commerce systems such as Amazon [5], travelling systems such as TravelJoy [6], movie-streaming platforms such as Netflix [7], and music applications [8]-[10] have achieved great success by making entertainment and shopping easily accessible and providing an amazing experience to users especially during the COVID-19 pandemic. Many recommendation system approaches have been proposed and developed in order to meet the growing needs of users and to overcome the encountered problems in the recommendation process. According to [2], three main types of recommendation systems have been proposed in the literature: collaborative filtering (CF) [11], recommendation systems based on the content [12], and hybrid recommendation systems [13], [14].

CF recognizes commonalities between users or between items on the basis of relevance indications [15]. A content-based recommendation system suggests those items that have similar features to items that the user has liked before [3]. A typical content-based recommendation first creates user profile using user feedback and ratings about items. A hybrid recommendation system combines multiple approaches together to achieve some synergy between them [16], [17]. CF approach is the most used approach in recommendation systems due to their efficiency and simplicity [18]. In this section, we briefly review the main CF algorithms reported in the literature. These algorithms are based on a simple intuition. They assume that good recommendations can be derived from users sharing the same interests and preferences. These preferences can be expressed in several ways, either by using ratings based on users' interests [19], or by deducing from users' behavior, tracking their purchase history and time spent on web content, which is known as implicit return [20]. They can be expressed in the form of a matrix called rating matrix [21]. This is the basis for creating effective prediction models and user profiles [22]. There are two main approaches to CF: memory-based and model-based.

- Memory-based CF recommendation approaches directly exploit users' preferences [23] (drawing similar relationships between the users or the items based on the user-item rating matrix). The techniques revealed in the memory-based approach are considered the first algorithm of CF [24]. The recommendations are easily applied to the ratings matrix. The techniques used in the memory-based CF make it possible to obtain the similarity to calculate the distance between two users (user-based approach), or between two items (item-based approach), according to the evaluations, in the rating matrix [25].
- Model-based approaches (also called collaborative neighborhood-based filtering) construct or learn models from collected notes based on machine learning techniques such as clustering techniques [26], dimensionality reduction approaches [27], support vector machines, and neural networks [28].

CF relies on the users' community in the system. Its main characteristic is the use of ratings obtained from users' recommended items. The principle is to filter the flow of items as rated by the other users' community. If an item has been deemed interesting by a user, it will be automatically recommended to users with similar views in the past. Hence, the objective of CF is to predict, for an unrated element, the evaluation that the target user might assign, based on the correlation between their own ratings and the ratings of other users who have similar interests and preferences.

Currently, CF has become the most widely-used approach [19], which motivates a significant number of researchers working on this issue. Neighborhood selection is one of the concepts used in CF. Among the research carried out on this topic, Zhang and Hurley [29] proposed grouping user profiles into clusters of similar articles and composing the list of recommendations that fit in well with each cluster [30], posited that clustering improves the performance of the recommendation. Adamopoulos [31] suggests a new neighborhood-based probabilistic approach as an improvement to the nearest k nearest standard algorithm.

In the next section, we review the memory-based techniques using user-based approach (UBCF). The top-K users who have similar preferences to a given user are called k-nearest neighbors (KNN). We use similarity measures to draw similarities or correlations among users to identify the KNN. In KNN, the value of K is the number of similar neighbors we need to predict ratings [23]. Despite its advantages, CF has a number of drawbacks affecting the accuracy of recommendations. Among these drawbacks is the gray sheep users, designating users who have unusual preferences and who do not share the same preferences with other users [15]. This problem makes the task of finding neighbors difficult.

This work aims to mitigate the gray sheep problem and to enhance the accuracy of recommendations based on the opposite preferences of users. The basic idea governing this lies in generating imaginary users based on dissimilar ones in order to enhance the user neighborhood. The underlying assumption of our approach is that if user X has an opposite preference to user Y, then the imaginary user  $\neg X$  has a similar preference to the user Y. Our approach increases the number of comparable neighbors, amplifies the density of the neighborhood, and then allows for building good recommendations.

The remainder of this paper is organized as follows: in section 2, we provide an overview of the basic approaches to CF. In section 3, we tackle the related work with a focus on the gray sheep users

problem. In section 4, we discuss our proposed approach and the novelty of this work. The experiments and results are presented in section 5, followed by the conclusion and future work in Section 6.

**2. BACKGROUND**

**2.1. Collaborative filtering recommendation process**

The memory-based approach is based on three steps presented in Figure 1 [32]. The first step before entering the collaborative filtering recommendation process consists in collecting users' data in need for recommendations. These data (rated films in this case) serve as a request for the algorithm. In the Figure 2 an example of data collection.



Figure 1. Collaborative filtering recommendation process



Figure 2. An example of data collection

**2.1.1. Data representation**

The second step of CF consists in constructing the evaluation matrix and filling in the empty values. In fact, in most cases, the scoring matrix is usually filled in because users do not score items regularly [27]. The most used technique in the CF is replacing the empty squares of the matrix with the average user ratings. In the Figure 3, a small-scale example of data representation.

User-Item Rating				
	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>
U <sub>1</sub>	5		3	
U <sub>2</sub>		4	2	5
U <sub>3</sub>		5	2	

Figure 3. A small-scale example of data representation

**2.1.2. Neighborhood formation**

In this step, we look for the neighborhood of the most similar users using a similarity metric. There are different measures for obtaining the similarity. However, the most extended ones are the Pearson correlation coefficient and cosine similarity [24]. This work employs two formulas: the Pearson correlation coefficient and the cosine. For the Pearson correlation coefficient, its values are between -1 and +1. It is considered a standard way to measure correlation [29]. However, the values obtained for the cosine similarity vary from 0 to 1 (0 means there is no correlation between the two users and 1 means that they are identical). Thus, we use two formulas to calculate the similarity between two users, a and b, Pearson correlation formula,

$$sim_{a,b} = \frac{\sum_{j=1}^n (r_{aj} - \bar{r}_a)(r_{bj} - \bar{r}_b)}{\sqrt{\sum_{j=1}^n (r_{aj} - \bar{r}_a)^2 \sum_{j=1}^n (r_{bj} - \bar{r}_b)^2}} \tag{1}$$

cosine similarity formula,

$$sim_{a,b} = \cos(a, b) = \frac{a \cdot b}{\|a\| \|b\|} = \frac{\sum_{j=1}^n r_{aj} \cdot r_{bj}}{\sqrt{\sum_{j=1}^n r_{aj}^2} \cdot \sqrt{\sum_{j=1}^n r_{bj}^2}} \tag{2}$$

**2.1.3. Predictions generation**

In this step, after selecting the nearest k-neighbors for the active user using the similarity degree based on the ratings matrix, the CF process generates predictions for unseen items. The prediction can be generated using the (3),

$$p_{s,i} = \bar{r}_s + \frac{\sum_{p=1}^k (r_{p,i} - \bar{r}_p) \cdot sim_{s,p}}{\sum_{p=1}^k |sim_{s,p}|} \tag{3}$$

in this formula, we calculate the predictions of rating for all the items that have not yet been seen by user s. We use the KNN technique where K represents the number of closest neighbors and  $\bar{r}_s$  represents the user's average rating s. The Figure 4 present a small-scale exemple of prediction' generation.

User-item rating				
	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>
U <sub>1</sub>	5	3	3	4
U <sub>2</sub>	3	4	2	5
U <sub>3</sub>	4	5	2	5

Figure 4. A small-scale exemple of predictions' generation

**2.2. Evaluation metrics**

After generating the predictions, we move to the stage of evaluating the performances of these predictions. In the literature, the performance of recommendation systems is measured with two commonly-used evaluation measures: Mean absolute error (MAE) and root mean squared error (RMSE). MAE calculates the mean absolute differences between the predicted values and the actual values as presented in (4),

$$MAE = \frac{\sum_{(s,i)} |p_{s,i} - r_{s,i}|}{N} \tag{4}$$

as N represents the number of the predicted rating calculated during the test phase,  $r_{s,i}$  is the actual rating given by the user s to the item i, and  $p_{s,i}$  is the rating predicted by the user s for the item i. RMSE is a standard way to measure the error of a model in predicting quantitative data. Formally, it is defined as (5),

$$RMSE = \sqrt{\frac{\sum_{(s,i)} (p_{s,i} - r_{s,i})^2}{N}} \tag{5}$$

in this section, we have cited the steps of the CF user-based approach. We can see that it is easy to implement and give good recommendations, but regardless of these advantages, there are many disadvantages to this approach that influence the results, such as the small quantity, the scalability, and the gray sheep problem. In the latter, it is difficult to find similar neighbors for a user's preference, which undermines the results obtained.

**3. RELATED WORK**

In this section, we discuss the problem of gray sheep users and deal with how this problem is overlooked in research on recommendation systems. Claypool *et al.* [33] confirms that the efficiency of traditional CF algorithm varies from one user to another. There are two main categories for users: White sheep (WS) and gray sheep (GS). WS users have high similarity to many other users (the correlation value is high), whereas the GS users are dissimilar or partly similar to other users and have a lower correlation coefficient with almost all users [31]. Therefore, the user recommendations become less accurate due to GS users [34]; hence, they do not benefit from recommendation systems. There are some works that deal with the problem of gray sheep users [35]-[37]. Claypool *et al.* [33] highlighted the problem of GS and onlined a hybrid recommendation system for updated recommendations. They combined CF and content-based

filtering approaches, using an average-weighted approach. However, they did not specifically target GS users nor did they offer a formal solution for this problem [38]. Used the MovieLens dataset to test this approach for a CF domain using. As this is a simulation, they did not describe a method for identifying these users and meeting their needs. GS users can be recognized using clustering algorithms offline, where the similarity threshold for separating these users from the rest of the clusters can be found empirically [37].

To identify GS users in the system, many approaches are suggested, including re-using outlier detection techniques based on user-user similarity distribution [39]. This is a distribution-based identification technique for GS users which borrows from the detection of outliers and the search for information, while taking into account the specifics of the preference data on which CF relies [23], i.e., clustering-based approaches [37], [40] or social network approaches [41]. In all, these approaches identify GS users with accuracy and eliminate GS users while making recommendations for the rest of the users with a high degree of accuracy. Hence, they do not consider GS users. In order to solve this problem, this work deploys all users and benefits from GS users.

#### 4. PROPOSED METHOD

As stated earlier, the basic CF approach uses the K-nearest neighbors to make new predictions. It only relies on users who have similar preferences to the active user, regardless of users with low similarity or dissimilarity in the prediction phase. In GS cases, the similarity between the active user and other users is still low or nonexistent, as most of them are distant. Figure 5 shows concretely the case of GS in the CF process. The three possible cases are delineated thus, Figure 5 shows an example of hypothetical neighbors that can be generated after inverting preferences. We have represented the new imaginary neighbors by red dots, formed via the step of increasing the matrix. The new neighbors can be positively correlated with the target user.

The core of our approach is to benefit users (GS users) whose preferences are different from the target user. The underlying assumption of our approach is that users must have more or less the same interests. If user X's interests are opposite to user Y, then the imaginary user  $\neg X$  would have the same interests as the user Y. Therefore, additional information will be provided to the recommendation engine to make good recommendations. The new process of our approach includes an additional step that increases the neighborhood number of the active user, which is called ratings matrix augmentation as shown in Figure 6.

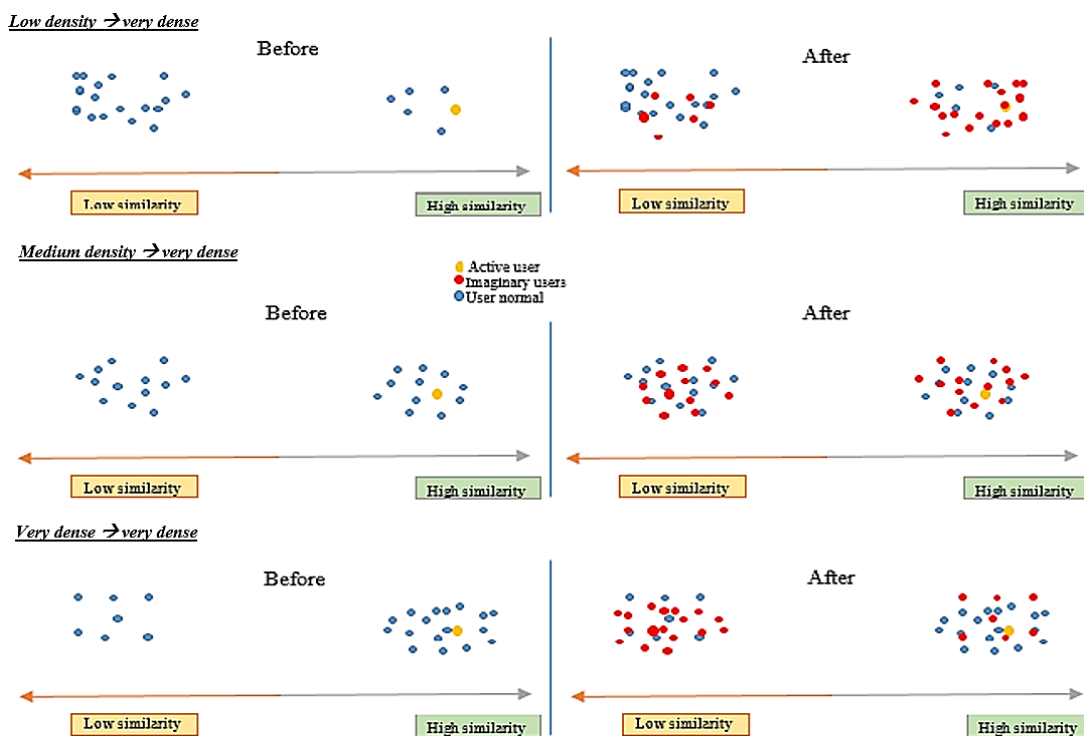


Figure 5. Example of GS cases in neighborhood-based techniques

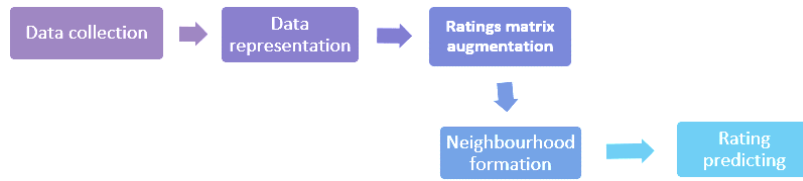


Figure 6. New memory-based CF process

The augmentation step of the ratings matrix consists of adding rows in the ratings matrix that represent opposed users to real users. The imaginary user is obtained by deducing the opposite preference from each item evaluated using (6),

$$\neg r_{uj} = Max - r_{uj} + Min \tag{6}$$

$r_{uj}$ : the rating of user  $u$  for an item  $j$ .  $Max$  and  $Min$ , the high and low values respectively in a given numerical scale. Example, by providing a sample of rating that ranges from 1 to 5 in the Figure 7 if a user  $u$  rates an item as  $r_{uj}= 5$ - the estimated rating of the user  $\neg u$  will be  $\neg r_{uj}= 1$ . Figure 7 illustrates a sample-case of imaginary neighbors that can be generated after inverting preferences. We represented the new imaginary neighbors by red dots, formed via the step of increasing the matrix. The new neighbors thus can be positively correlated with the target user. Figure 7 shows an example of an opposite user on a 5-point scale using the formula 6 above.  $U$  represents the opposite user of the user  $U$  after the application of the formula. It relies on the inference of an opposing user's ratings by providing the opposite preference of a given user. Finally, we list the pseudonyms of our proposed method in algorithm matrix augmentation as shown in Figure 8.

### 5. RESULT AND DISCUSSION

In this section, many experiments are performed to demonstrate the novelty and efficiency of our approach. Therefore, we divided our dataset into 80% for the training set and 20% for the test set. We calculated the means of the results of a cross-validation of 10 times. We also implemented a system of film recommendation under R thanks to Recommend erlab [42] with the MovieLens and FilmTrust datasets. The objective is to check the performance of our proposed approach (AUBCF) with the traditional user-based CF approach (UBCF) using real-world datasets. A brief description of the datasets used will be in order, ensued by the evaluation procedure, and the specification test environment. Hence, the results were acquired from comparisons to come up the most successful approach.

Items \ Users	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>	I <sub>5</sub>	I <sub>6</sub>	I <sub>7</sub>
$U$	4		2		5		3
$\neg U$	2		4		1		3

Figure 7. Example of an opposite rating matrix on a 5-point scale

```

Algorithm : Matrix Augmentation
Input : Matrix T[nbrusers][nbritems] ;
Output: MatrixAugmentation TA [2 x nbrusers][nbritems];
Initialize Max=5, Min=1 ;
cpt1 =1
cpt2 =1
while cpt1 <- nbrusers do // for all users in the dataset
  while cpt2 <- nbritems do // for all his/her rated items
    if T[cpt1][cpt2]>0
      TA[nbrusers+cpt1][cpt2]= Max - T[cpt1][cpt2] +Min
    cpt2++
  cpt1++
return TA
  
```

Figure 8. Algorithm matrix augmentation

### 5.1. Datasets collection

We executed our experiments with two commonly-used datasets: FilmTrust and MovieLens. Both are academic research projects of web-based movie recommendation systems. MovieLens is a set of evaluation data on a scale of 5 points ranging from 1 (bad) to 5 (excellent). It includes 1682 films, 943 users, and 100,000 rankings. The FilmTrust dataset includes 1856 users, 2092 movies, and 759922 reviews. It was collected from a social network based on a video recommendation system including reviews. The odds are 5-point scales ranging between 0.5 and 4 stars.

### 5.2. Experiments

All the experiments and techniques were performed on Intel i5 at 2.4 GHz and 8 GB RAM, using MovieLens and FilmTrust datasets. The experimental evaluation of our suggested method is carried out in this section. And the results are based on a variety of frequently used metrics with various parameters.

#### 5.2.1. Pearson correlation

The Figures 9 and 10 shows the results obtained by comparing our proposed approach named AUBCF (UBCF augmented) and the user-based CF approach (UBCF) as a basic approach for the FilmTrust dataset. Figure 9 represents a comparison of MAE where the horizontal axis is the size of the neighborhood used for the calculation of MAE. The figure shows that our approach (AUBCF) decreases regularly for the MAE, while the traditional approach (UBCF) decreases to N=40 and then remains stable until N=60 where the MAE begins to increase. In Figure 10, we can see that the MAE of our approach (AUBCF), in green, and the traditional approach (UBCF), in red, are inversely proportional to the number of users in the neighborhood. The traditional approach (UBCF) has a higher MAE than our approach (AUBCF).

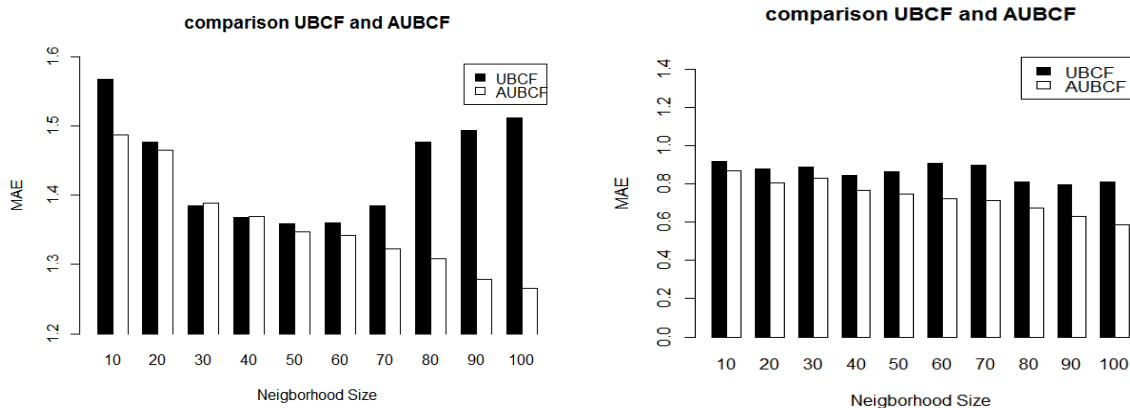


Figure 9. MAE comparison using FilmTrust dataset    Figure 10. MAE comparison using MovieLens dataset

#### 5.2.2. Cosine

Figures 11 and 12 demonstrate the results obtained by comparing the traditional approach (UBCF), in red, and the proposed approach (AUBCF), in green, for each dataset. The diagram represents a comparison of MAE where the horizontal axis is the size of the neighborhood in each experiment. It increases from 10 to 100 at the interval of 10. The proposed approach (AUBCF) in Figure 11 remains regular for the MAE while the traditional approach (UBCF) increases to N=30; then it remains stable until N=60 where the MAE begins to decrease. In Figure 12, the MAE of proposed approach (AUBCF) and the traditional approach (UBCF) are inversely proportional to the size of the neighborhood. They decrease steadily up to N=60, then they remain stable up to N=100. Thus, our approach (AUBCF) has a lower MAE than the traditional approach (UBCF). All in all, we conclude from these experiments that the proposed approach (AUBCF) offers better performance than the traditional approach (UBCF) in both datasets.

### 5.3. Statistical inference

In most experiments, it is important to make sure that the observed difference between the proposed method and baseline one is statistically significant; and it is unlikely to be due to chance or noise in the data. the appropriate statistical test to use is the Wilcoxon test. The Wilcoxon test is a nonparametric statistical test that compares two data samples without assuming the data to have a specific distribution, the goal of the test is to decide whether the population distributions are identical or not. Our null hypothesis is that the results of algorithm AUBCF and the results of algorithm UBCF are identical populations, that any small gain, or loss,

observed is not statistically significant. Generally, we reject the null hypothesis if the p-value is less than a certain threshold (often 0.05). In other words, if  $p\text{-value} < 0.05$  we can infer that the difference is statistically significant. Comparison of p-value between Pearson correlation and cosine by Wilcoxon test for both datasets as shown in Table 1. According to Table 1 all p-values are the threshold (0.05), we rejected the null hypothesis and we can say that the difference is statistically significant. Finally, the obtained results of algorithm AUBCF are better than the results of algorithm UBCF.

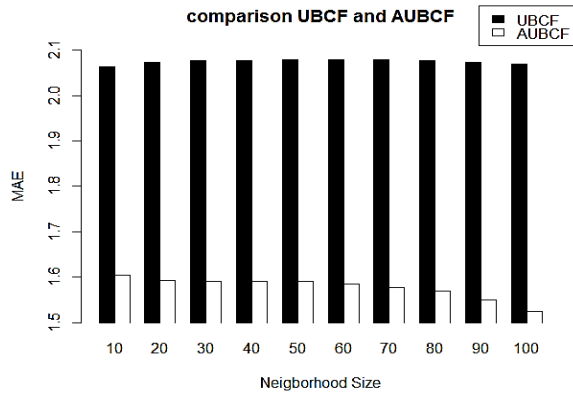


Figure 11. MAE comparison using FilmTrust dataset

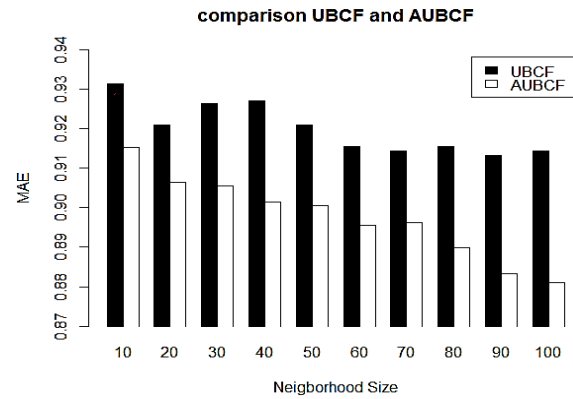


Figure 12. MAE comparison using MovieLens dataset

Table 1. Comparison of p-value between Pearson correlation and cosine by Wilcoxon test for both datasets

	Pearson correlation	Cosine
FilmTrust	6.900406e-25	0.01639486
MovieLens	4.473234e-05	0.0002432247

## 6. CONCLUSION & FUTURE WORKS

Despite the popularity and the great usage of CF, it is not without limitations as it still has to overcome the GS problem. Therefore, this work proposed a new CF approach to solve this problem. This approach aims to increase the number of neighbors for the active user based on users with different interests and preferences. To evaluate our algorithm, we compared it to UBCF as a traditional approach. The comparison was done on two datasets, FilmTrust and MovieLens. The obtained results show that our approach outperforms UBCF and improves prediction accuracy for GS problems.

The contribution of our work can be summarized in a three main points. First, our approach makes full use of the rating data to improve the accuracy of recommendation systems. All the rating data from users are used in the model, not just the WS users rating data. Second, the problem of GS users is solved in our model which makes it possible to obtain an accurate similarity when there is no correlation between two users. Third, this paper proposes a new approach for collaborative filtering which shows superior performance than the traditional collaborative filtering. In future work, we will consider the idea of hybridizing our approach with various machine learning techniques in order to improve the accuracy of the recommendations.

## REFERENCES

- [1] N. Polatidis and C. K. Georgiadis, "A dynamic multi-level collaborative filtering method for improved recommendations," *Comput. Stand. Interfaces*, vol. 51, pp. 14-21, 2017, doi: 10.1016/j.csi.2016.10.014.
- [2] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 6, pp. 734-749, 2005, doi: 10.1109/TKDE.2005.99.
- [3] G. Takács and D. Tikk, "Alternating least squares for personalized ranking," *RecSys'12 - Proc. 6th ACM Conf. Recomm. Syst.*, pp. 83-90, 2012, doi: 10.1145/2365952.2365972.
- [4] F. Ortega, B. Zhu, J. Bobadilla, and A. Hernando, "Knowledge-Based Systems CF4J: Collaborative filtering for Java," *Knowledge-Based Syst.*, pp. 1-6, 2018, doi: 10.1016/j.knosys.2018.04.008.
- [5] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, no. 1, pp. 76-80, 2003, doi: 10.1109/MIC.2003.1167344.
- [6] K. A. F. A. Samah, N. Sabri, R. Hamzah, R. Roslan, N. A. Mangshor, and A. A. M. Asri, "Brute force algorithm implementation for traveljoy travelling recommendation system," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 16, no. 2, pp. 1042-1049, 2019, doi: 10.11591/ijeecs.v16.i2.pp1042-1049.
- [7] C. A. Gomez-uribe and N. Hunt, "The Netflix Recommender System Algorithms, Business Value.pdf," vol. 6, no. 4, 2015, doi: 10.1145/2843948.






- [8] V. Kasinathan, A. Mustapha, T. S. Tong, M. F. C. A. Rani, and N. A. A. Rahman, "Heartbeats: Music recommendation system with fuzzy inference engine," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 16, no. 1, pp. 275-282, 2019, doi: 10.11591/ijeecs.v16.i1.pp275-282.
- [9] O. Celma, "Music recommendation and discovery in the long tail," *Citeulikeorg*, p. 252, 2008, [Online]. Available: <http://www.citeulike.org/user/bemike/article/4099800>
- [10] A. F. Smeaton and J. Callan, "Personalisation and Recommender Systems in Digital Libraries Joint NSF-EU DELOS Working Group Report," *Library (Lond.)*, vol. 5, no. May, pp. 299-308, 2003, doi: 10.1007/s00799-004-0100-1.
- [11] Y. El M. El Alami, E. H. Nfaoui, and O. El Beqqali, "A novel collaborative filtering approach based on social network experts," *2016 Int. Conf. Inf. Technol. Organ. Dev. IT4OD 2016*, pp. 1-7, 2016, doi: 10.1109/IT4OD.2016.7479277.
- [12] M. J. Pazzani and D. Billsus, "Content-Based Recommendation Systems," *Constr. Build. Mater.*, vol. 171, pp. 546-557, 2007, doi: 10.1016/j.conbuildmat.2018.03.149.
- [13] R. Burke, "Hybrid Recommender Systems : Survey and," *User Model. UserAdapted Interact.*, vol. 12, no. 4, pp. 331-370, 2002, doi: 10.1023/A:1021240730564.
- [14] Y. El Madani El Alami, E. H. Nfaoui, and O. El Beqqali, "Toward an effective hybrid collaborative filtering: A new approach based on matrix factorization and heuristic-based neighborhood," *2015 Intell. Syst. Comput. Vision, ISCV 2015*, 2015, doi: 10.1109/ISACV.2015.7105543.
- [15] R. Burke, "Evaluating the dynamic properties of recommendation algorithms," *RecSys'10 - Proc. 4th ACM Conf. Recomm. Syst.*, pp. 225-228, 2010, doi: 10.1145/1864708.1864753.
- [16] D. K. Behera, M. Das, S. Swetanisha, and P. K. Sethy, "Hybrid model for movie recommendation system using content K-nearest neighbors and restricted boltzmann machine," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 23, no. 1, pp. 445-452, 2021, doi: 10.11591/ijeecs.v23.i1.pp445-452.
- [17] L. N. Tondji, "Web Recommender System for Job Seeking and Recruiting," no. February, 2018, doi: 10.13140/RG.2.2.26177.61286.
- [18] M. D. Ekstrand, "Collaborative Filtering Recommender Systems," *Found. Trends@ Human-Computer Interact.*, vol. 4, no. 2, pp. 81-173, 2011, doi: 10.1561/1100000009.
- [19] G. Jawaheer, M. Szomszor, and P. Kostkova, "Comparison of implicit and explicit feedback from an online music recommendation service," pp. 47-51, 2006, doi: 10.1145/1869446.1869453.
- [20] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative Filtering for Implicit Feedback Datasets," *Gastroenterology*, vol. 1, p. S415, 2008, doi: 10.1109/ICDM.2008.22.
- [21] A. Hernando, J. Bobadilla, and F. Ortega, "A non negative matrix factorization for collaborative filtering recommender systems based on a Bayesian probabilistic model," *Knowledge-Based Syst.*, vol. 97, pp. 188-202, 2016, doi: 10.1016/j.knsys.2015.12.018.
- [22] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, "Recommendation systems: Principles, methods and evaluation," *Egypt. Informatics J.*, vol. 16, no. 3, pp. 261-273, 2015, doi: 10.1016/j.eij.2015.06.005.
- [23] B. Gras, A. Brun, and A. Boyer, "Identifying Grey Sheep Users in Collaborative Filtering," pp. 17-26, 2016, doi: 10.1145/2930238.2930242.
- [24] J. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," *Proc. 14th Annu. Conf. Uncertain. Artif. Intell.*, 1998, pp. 43-52, doi: 10.1111/j.1553-2712.2011.01172.x.
- [25] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," *Proc. 10th international conference on World Wide Web*, vol. 1, pp. 285-295, 2001, doi: 10.1145/371920.372071.
- [26] C. F. Tsai and C. Hung, "Cluster ensembles in collaborative filtering recommendation," *Appl. Soft Comput. J.*, vol. 12, no. 4, pp. 1417-1425, 2012, doi: 10.1016/j.asoc.2011.11.016.
- [27] A. Paterk, "Improving regularized singular value decomposition for collaborative filtering," *Proc. KDD cup Work.*, pp. 2-5, 2007, doi: 10.1145/1557019.1557072.
- [28] S. Agrawal and J. Agrawal, "Survey on anomaly detection using data mining techniques," *Procedia Comput. Sci.*, vol. 60, no. 1, pp. 708-713, 2015, doi: 10.1016/j.procs.2015.08.220.
- [29] M. Zhang and N. Hurley, "Novel item recommendation by user profile partitioning," *Proc. - 2009 IEEE/WIC/ACM Int. Conf. Web Intell. WI 2009*, vol. 1, pp. 508-515, 2009, doi: 10.1109/WI-IAT.2009.85.
- [30] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Recommender systems for large-scale e-commerce: Scalable neighborhood ...," *Proc. Fifth ...*, 2002, [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.4.6985&rep=rep1&type=pdf>
- [31] P. Adamopoulos, "On discovering non-obvious recommendations: Using unexpectedness and neighborhood selection methods in collaborative filtering systems," *WSDM 2014 - Proc. 7th ACM Int. Conf. Web Search Data Min.*, pp. 655-659, 2014, doi: 10.1145/2556195.2556204.
- [32] S. Bhaidani, "Recommender system algorithms," *Proc. Int. Conf. weblogs Soc. media ICWSM 2007*, 2008, [Online]. Available: <http://www.mie.utoronto.ca/undergrad/thesis-catalog/files/146.pdf>
- [33] M. Claypool, A. Gokhale, T. Miranda, P. Murnikov, D. Netes, and M. Sartin, "Combining content-based and collaborative filters in an online newspaper," *Proc. ACM SIGIR '99 Work. Recomm. Syst. Algorithms Eval.*, no. June, p., 1999.
- [34] A. Srivastava, P. K. Bala, and B. Kumar, "New perspectives on gray sheep behavior in E-commerce recommendations," *J. Retail. Consum. Serv.*, vol. 53, no. September 2018, p. 101764, 2020, doi: 10.1016/j.jretconser.2019.02.018.
- [35] S. Ghorbani and A. H. Novin, "An Introduction on Separating Gray-Sheep Users in Personalized Recommender Systems Using Clustering Solution," *Int. J. Comput. Sci. Softw. Eng.*, vol. 5, no. 2, pp. 14-18, 2016.
- [36] H. Jindal, S. Agarwal, and N. Sardana, "PowKMeans: A Hybrid Approach for Gray Sheep Users Detection and Their Recommendations," *Int. J. Inf. Technol. Web Eng.*, vol. 13, no. 2, pp. 56-69, 2018, doi: 10.4018/IJITWE.2018040106.
- [37] M. A. Ghazanfar and A. Prugel-bennett, "Fulfilling the Needs of Gray-Sheep Users in Recommender Systems, A Clustering Solution," *2011 International Conference on Information Systems and Computational Intelligence*, Harbin, China. 18 - 20 Jan 2011.
- [38] J. Van Diggelen, V. Dignum, L. Van Elst, and A. Abecker, "A decentralized approach for establishing a shared communication vocabulary," *Proceedings of the AAMAS 2005 Workshop on Agent Mediated Knowledge Management*, no. July, pp. 1-13, 2005.
- [39] Y. Zheng, M. Agnani, and M. Singh, "Identifying grey sheep users by the distribution of user similarities in collaborative filtering," *RIIT 2017 - Proc. 6th Annu. Conf. Res. Inf. Technol.*, pp. 1-6, 2017, doi: 10.1145/3125649.3125651.
- [40] M. A. Ghazanfar and A. Prügel-Bennett, "Leveraging clustering approaches to solve the gray-sheep users problem in recommender systems," *Expert Syst. Appl.*, vol. 41, no. 7, pp. 3261-3275, 2014, doi: 10.1016/j.eswa.2013.11.010.
- [41] K. Minsung and Il Im, "Resolving the 'Gray sheep' Problem Using Social Network Analysis (SNA ) in Collaborative Filtering (CF) Recommender Systems," 2014, vol. 20, no. 2, pp. 137-148, doi: 10.13088/jiis.2014.20.2.137.




- [42] M. Hahsler, "recommenderlab: A Framework for Developing and Testing Recommendation Algorithms," *Nov*, no. December 2013, pp. 1-37, 2011, [Online]. Available: <http://cran.r-project.org/web/packages/recommenderlab/vignettes/recommenderlab.pdf>

## BIOGRAPHIES OF AUTHORS






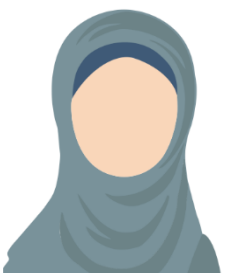
**Abdellah El Fazziki**    He received a B.S. and a M.S from University of Sidi Mohamed Ben Abdellah, Fez, Morocco in 2007, and 2009 respectively. He is currently a PHD student at University of Sidi Mohamed Ben Abdellah, Fez, Morocco. His main research interests include machine learning and recommendation systems. He can be contacted at email: [elf.abdellah@gmail.com](mailto:elf.abdellah@gmail.com).






**Yasser El Madani El Alami**    is an assistant Professor in the Department of Computer Science and at ENSIAS, University of Mohammed V in Rabat, Morocco, where he has been since 2018. He received a B.S. and a M.S from University of Sidi Mohamed Ben Abdellah, Fez, Morocco in 2005, and 2009 respectively. He received his Ph.D. in Computer Science from the University of Sidi Mohamed Ben Abdellah, in 2015. His research interests include recommender systems, information retrieval, E-learning and social network analysis. He can be contacted at email [y.alami@um5s.net.ma](mailto:y.alami@um5s.net.ma).






**Jalil Elhassouni**    received a PhD in computer sciences from University of Mohammed V in Rabat, Morocco in 2021. He is currently Head of the Infrastructure, network and security service at the Casablanca Urbain Agency. His research interests include the artificial intelligence, knowledge graphs, ontology modeling, ontology alignment, data integration, knowledge representation and semantic web. He can be contacted at email: [jalil.elhassouni@gmail.com](mailto:jalil.elhassouni@gmail.com).



**Ouafae El Aissaoui**    She got her Ph.D. in 2021 with a thesis in the educational data mining (EDM) field. Her research focus is the application of EDM techniques on educational data to reveal the learner's features (behaviors and preferences) that can be used to model the learner profile. Since her subscription in the PhD degree, she has succeeded in coming up with different contributions in this context using different machine learning techniques such as clustering, classification, and regression. She can be contacted at email: [ouafae.elaissaoui@usmba.ac.ma](mailto:ouafae.elaissaoui@usmba.ac.ma).



**Mohammed Benbrahim**    received the B.Eng. degree in electromechanical engineering from Higher National School of Mines, in 1997 and the M.Sc. and Ph.D. degrees in automatic and industrial informatics from Mohammadia School of Engineers, in 2000 and 2007, respectively. Currently, he is an Associate Professor at the Department of Physics and the program coordinator for the Master Smart Industry at Faculty of Sciences, Sidi Mohamed Ben Abdellah University. His research interests include robotics, automatic control, intelligent systems, predictive maintenance, modeling and optimization. He can be contacted at email: [mohammed.benbrahim@usmba.ac.ma](mailto:mohammed.benbrahim@usmba.ac.ma).