A highly scalable CF recommendation system using ontology and SVD-based incremental approach

Sajida Mhammedi[1], Noreddine Gherabi[1], Hakim El Massari[1], Zineb Sabouri[1], Mohamed Amnai[2]
[1]LaSTI Laboratory, National School of Applied Sciences, Sultan Moulay Slimane University, Beni-Mellal, Morocco
[2]Laboratory of Computer Sciences Research, Faculty of Sciences, Ibn Tofail University Kenitra, Kenitra, Morocco

Article Info

Article history:

Received Mar 23, 2023
Revised May 3. 2023
Accepted Jun 5, 2023

Keywords:

Accuracy
Collaborative filtering
Dimensionality reduction
Ontology
Scalability
Semantic filtering
Sparsity

ABSTRACT

In recent years, the need of recommender systems has increased to enhance user engagement, provide personalized services, and increase revenue, especially in the online shopping industry where vast amounts of customer data are generated. Collaborative filtering (CF) is the most widely used and effective approach for generating appropriate recommendations. However, the current CF approach has limitations in addressing common recommendation problems such as data inaccuracy recommendations, sparsity, scalability, and significant errors in prediction. To overcome these challenges, this study proposes a novel hybrid CF method for movie recommendations that combines the incremental singular value decomposition approach with an item-based ontological semantic filtering approach in two phases, online and offline. The ontology-based technique is leveraged to enhance the accuracy of predictions and recommendations. Evaluating our method on a real-world movie recommendation dataset using precision, F1 scores, and mean absolute error (MAE) demonstrates that our system generates accurate predictions while addressing sparsity and scalability issues in recommendation system. Additionally, our method has the advantage of reduced running time.

Corresponding Author:

Sajida Mhammedi
LaSTI Laboratory, National School of Applied Sciences, Sultan Moulay Slimane University
Beni-Mellal, Morocco
Email: sajida.mhammedi@usms.ac.ma

1. INTRODUCTION

A recommendation system is an intelligent system that captures user behavior on internet portals in order to forecast user interest in future online product purchases, movie viewing, or music listening. Because of the growing popularity of the internet market, several websites now provide numerous choices to their customers. Looking for the appropriate item in which the customer is interested among thousands of items in a short amount of time has become quite difficult. Thus, a recommendation system has been introduced to meet this challenging issue. A recommendation system may create data based on a user's previous purchase history, search patterns, and online behavior. When a new item or user enters the online website, the recommender system search for the existing data and recommends the best item based on the customer's preferences [1].

Recommendation systems, based on their functioning behavior, are categorized into three types of recommendation systems to create the most efficient suggestion: content-based recommendation systems, collaborative recommendation systems, and hybrid recommendation systems. Approaches such as collaborative filtering (CF) and content-based filtering (CBF) have mainly been developed to acquire insight into user preferences [2], [3]. Those preferences suggested in the CBF technique depend on their content similarity to items previously scored by the user. While the CF technique takes advantage of the similarity of users' tastes for suggestions [4], [5]. The CF technique is classified into two types: user-based and item-based. The similarity between users in the user-based CF method is estimated according to co-rated items. In contrast, item-based CF assesses this similarity among items instead of users. Items that users have previously admired will pique their attention. Compared to different techniques, the hybrid strategy combines the two filtering procedures and has superior prediction accuracy [6].

Even though CF has drawn attention owing to its effectiveness and simplicity [4], it still faces the following challenges: data sparsity [7], computation time, accuracy of recommendations, scalability, and data volume. To address the issues presented by the CF method, the hybrid recommendation strategy employs several information filtering techniques. The hybrid filtering approach is designed to produce more efficient and accurate recommendations than a single technique. In addition, the hybrid model overcomes the disadvantages of a single system by combining many techniques. In this research, we offer a recommendation approach that uses both ontological semantic filtering and an incremental algorithm to give high scalability when dealing with massive increases in user and item matrix sizes as well as sparsity issues. In order to have an accurate prediction with a decreased running time of the recommendations system. The rest of the paper is organized as: section 2 examines the relevant works to this research work, the proposed approach is described in section 3, section 4 discusses the practical implementation and evaluation of the proposed approach, and followed by the final section 5 which concludes the paper.

2. RELATED WORK

Several strategies for recommendation systems have been established in prior research. Nowadays, recommender systems are essential to speed up internet users' searches for relevant content. In the area of recommender systems, using ontologies as a knowledge base is becoming increasingly popular in modeling tasks, inferring new knowledge [8], [9] or computing similarity for recommender systems. Adopting ontologies in information systems intends to model information at the semantic level by structuring and organizing a set of hierarchical terms or concepts within a domain and modeling the relationships between these sets of terms or concepts using a relational descriptor [10], [11]. Recommender systems based on knowledge represented by ontologies are then proposed by explicitly soliciting user requirements for these elements and an in-depth understanding of the underlying domain for similarity measures and prediction computation. In relation to the significant number of published studies [11], enhance user profile representation by implementing an ontology-based recommendation system. By introducing domain ontologies into the system, the suggested technique is able to uncover relationships between users and their favorite choices regarding items. The authors developed several experiments based on offline tests. They also compared the new recommendation approach to collaborative approaches. To improve the quality of the recommendations, Hassan *et al.* [12] used item semantic knowledge. As a result, they created a hybrid semantic improved recommendation strategy that combines the inferential ontology-based semantic similarity (IOBSS) with the classic item-based CF method. Kermany and Alizadeh [13] suggested multi-criteria recommender systems using adaptive neuro-fuzzy inference system (ANFIS) relies on ontological item-based and user demographic information. Their method was tested using Yahoo movies platform dataset. Moreover, according to their results, the accuracy of multi-criteria recommendation system can be increased by incorporating semantic information.

Dimensionality reduction techniques have been widely used in the literature of recommendation systems. Among the most successful is the dimension reduction method called singular value decomposition (SVD) and its variants and principal component analysis (PCA) [14], [15]. These techniques are used to reduce the dimensionality of the data, which helps in handling the sparsity of the data and improving the efficiency and accuracy of the recommendation process. The literature shows that these techniques have been successful in improving the recommendation performance on various datasets, especially after the challenge launched by Netflix. Indeed, many works like [16] analyzing the results of the challenge demonstrated the superiority in terms of accuracy of approaches after applying dimensionality reduction techniques over CF algorithms. Recent research related to our works has also used SVD as a technique in CF for recommendation systems. For instance, Wang *et al.* [17] proposed a CF algorithm that incorporates trust between users to improve recommendation accuracy. The algorithm combines the traditional SVD method with a trust factor matrix, the results show that it outperforms other state-of-the-art CF methods in terms of recommendation accuracy. Nilashi *et al.* [18] combine CF with ontology-based techniques and dimensionality reduction. The proposed recommender system uses ontology and dimensionality reduction techniques to improve the accuracy and coverage of CF. It combines semantic similarity and matrix factorization to handle the sparsity problem and provide more personalized recommendations. However, the use of incremental SVD has been proposed as a way to improve scalability and performance compared to non-incremental SVD [19]. Brand [20] uses an incremental SVD approach with incomplete data to solve the issue of uncertain new data with missing values and/or affected by correlated noise. In comparison to SVD technique, using incremental SVD in recommendation systems will updates the factorization model using only new information instead of recomputing the entire model from scratch, which can be computationally expensive and time-consuming for large datasets. As a result, incremental SVD can reduce the training time and improve the efficiency of the recommendation system without sacrificing accuracy. Overall, our contribution lies in presenting a comprehensive recommendation method that combines dimensionality reduction, ontology-based techniques, and incremental SVD to address key challenges in recommendation systems. By leveraging these techniques, we aim to improve recommendation accuracy, scalability, and efficiency which will ultimately enhancing the user experience and satisfaction.

3. HYBRID RECOMMENDER SYSTEM PROPOSITION

Figure 1 shows the diagram illustrating how the proposed recommendation system works. The suggested recommendation system aims to provide efficient, scalable, and accurate recommendations. Two significant aspects to examine in the suggested system process. In the first phase, several tasks are performed during the construction of the recommendation model, such as clustering of items and users based on rating, dimensionality reduction using the SVD algorithm, and constructing item-user similarity matrices. First, the system is supplied with a user-item matrix that specifies the user's rating given to each item. As a result, item clusters must be constructed using fuzzy c means clustering to determine the similarity between items. The pairwise similarity between them is computed to regroup items based on similarity. The overall similarity is obtained by calculating the item-based and ontology-based similarity averages.

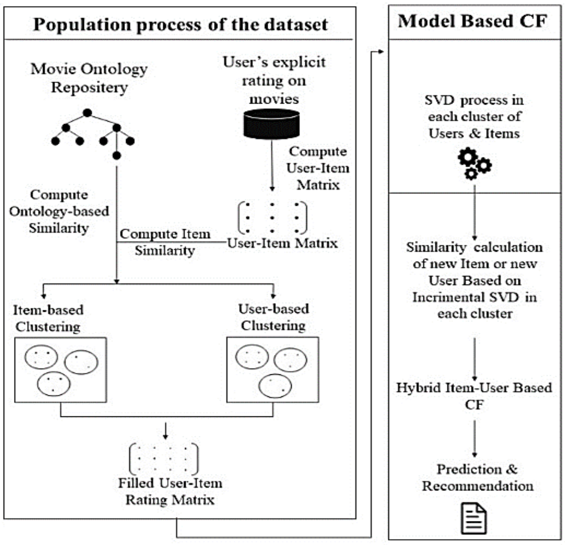

Population process of the dataset
Movie Ontology Repository
User's explicit rating on movies
Compute User-Item Matrix
Compute Ontology-based Similarity
Compute Item Similarity
User-Item Matrix
Item-based Clustering
User-based Clustering
Filled User-Item Rating Matrix
Model Based CF
SVD process in each cluster of Users & Items
Similarity calculation of new Item or new User Based on Incrimental SVD in each cluster
Hybrid Item-User Based CF
Prediction & Recommendation

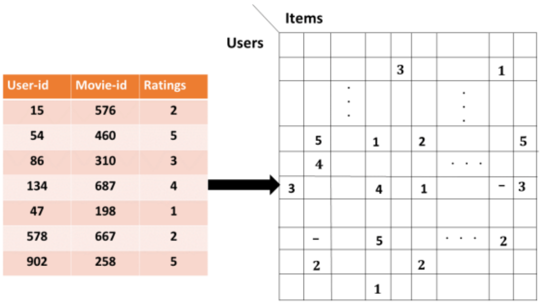
Figure 1. Proposed system framework using ontology and incremental SVD

A new algorithm based on ontologies is suggested to compute the item similarity. Following that, we created decomposition matrices using SVD for the user-item cluster. It is worth mentioning that we are working on the SVD model for items and users. As a result, in each matrix, similarity computation is correctly performed after the matrix decomposition process. After the comparable item clusters have been produced, it is proposed to predict a rating for the current user who has yet to rate every item in the system to eliminate sparsity in the user-item matrix. The incremental SVD is employed as part of the recommendation process' second phase (online phase) to predict and recommend tasks for targeted users and items. We follow the same procedure as the item-based suggestion. Finally, in a meaningful way, integrate user-and item-based predictions. In the following subsections, the approach is discussed in depth.

3.1. Preprocessing of data

The initial step in our research is to preprocess the dataset in order to make it suitable for the proposed method. This involves conducting the necessary preparation processes that real-life data typically require for analysis. In our approach, we begin by transforming movie ratings into a user-item matrix, often referred to as a rating utility matrix. This matrix captures the ratings provided by users for different movies in

Figure 2. However, this matrix is typically sparse, meaning many cells are empty as they represent movies the user has not rated. CF algorithms typically work with dense matrices, so we need to convert the sparse matrix into a dense matrix by applying normalization techniques. The empty cells in the matrix correspond to new users, new movies, or movies not rated by anyone. Users who have expressed positive sentiment (indicating user preference) towards a movie are assigned ratings of 4 or 5, while users who have shown negative sentiment (indicating user disinterest) are assigned ratings of 1 or 2. Therefore, to address item and user bias in the ratings, we normalize the ratings using mean normalization.



User-id	Movie-id	Ratings
15	576	2
54	460	5
86	310	3
134	687	4
47	198	1
578	667	2
902	258	5

Figure 2. Transform original data to user-item rating matrix

3.2. Movie ontology

In this research, we use the movie ontology (MO) created based on the ontology web language (OWL) standard and at the Department of Informatics in the University of Zurich [21]. MO elucidates the semantic ideas and concepts related to the domains of the films. The class "movie" is the main class and all movies are considered instances of it. Many research have demonstrated that using an ontology-based semantic approach improves the prediction accuracy of recommendation systems [22], [23].

3.3. User-based clustering

In user clustering, users are grouped based on similar preferences, as determined by their ratings. After clustering the users, each cluster's views aggregate is utilized to predict unidentified ratings for target users or predict which items they like or dislike. Since clusters contain a restricted number of users, there is no need to evaluate all users. Thus, it results in improving performance.

3.4. Item-based clustering

3.4.1. Compute ontology-based item similarity

Ontologies supply immense knowledge on any topic, which might be highly valuable in the recommendation system [24]. Most studies ignored ontologies' multilevel and complicated structures and used just one feature to determine item similarity according to ontology. For instance, several researchers have relied only on a movie's "genre" to identify a related collection of films based on ontology. In the context of a movie recommendation system, let's consider Figure 3. In this example, we can assume that CL represents the movie class. Within this movie class, we have two attributes: At1 and At2, which could represent characteristics such as the release date and copyright information. Additionally, we introduce a subclass called SCl, which represents the "movie origin". This subclass includes attributes At3, At4, and At5, which correspond to specific regions such as North Africa, Asia, and Europe. By organizing the movie data in this hierarchical manner, we can capture more detailed information about movies and their origins. This ontology-based approach allows us to categorize and represent movies based on their attributes, enabling more sophisticated recommendation algorithms to provide personalized movie suggestions to users. This work uses the binary Jaccard similarity coefficient to compute item-based semantic similarity. For two items to be similar, their attributes and the attributes of their subclasses must be similar [25]. The average of the values is determined using recursive computing to find the similarity between items until the maximum depth defined at the beginning is reached. As a result, in (1) to (3) is used to determine ontology-based similarity:

$$Sim\left(Cl_i \cdot At, Cl_j \cdot At\right) = \frac{A_{11}}{A_{11}+A_{01}+A_{10}} \tag{1}$$

$$Sim_{\text{ontology}}\left(It_i, It_j\right) = \frac{\sum_{k=1}^{n} Sim\left(Cl_i \cdot At(z), Cl_j \cdot At(z)\right)}{n} \tag{2}$$

$$Sim_{\text{ontology}}\left(It_i, It_j\right) = \frac{\sum_{k=1}^{n} \text{Sim}\left(Cl_i \cdot At(z), Cl_j \cdot At(z)\right) + \sum_{p=1}^{m} \text{Sim}\left(\text{SCl}_i \cdot At(d), \text{SCl}_j \cdot At(d)\right)/m}{n} \tag{3}$$

Assuming no attribute in the ontology is a subclass, in (2) is the ontology-based similarity between two items It_i *and* It_j. In (1) is the semantic similarity between classes Cl_i and Cl_j of two items It_i and It_j for a specific attribute At, the total number of attributes is represented by n. In (3) computed if attributes, subclass with its attributes exist in the ontology. To determine the ontology-similarity between two objects specified by the domain's common. It requires the following two inputs: i) ontology of items with classes, properties, and relationships; and ii) I represent the set of all items. The semantic similarity matrix (SSM) is computed, which measures semantic similarity between two items based on ontology.

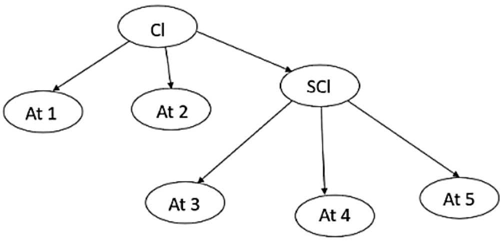


Figure 3. Example of item's ontology

3.4.2. Calculation of item similarity using explicit user ratings

The similarity between items is determined based on explicit ratings supplied by users in the user-item rating matrix, where I represent the set of items, U represents the set of users, and Rui represents the score given by user u to item I, as seen in Figure 4. The similarity metric that was used in (4):

$$Sim\left(It_i, It_j\right) = \sqrt{\sum_{u=1}^{n} \frac{\left(It_{iu} - It_{ju}\right)^2}{\left(It_{iu}\right)^2 - \left(It_{ju}\right)^2}} \tag{4}$$

where It_{iu} and It_{ju} represent the values of the ratings given by user u to item It_i and item It_j, respectively. 1≤u≤l, both items were rated by the total number L of users.

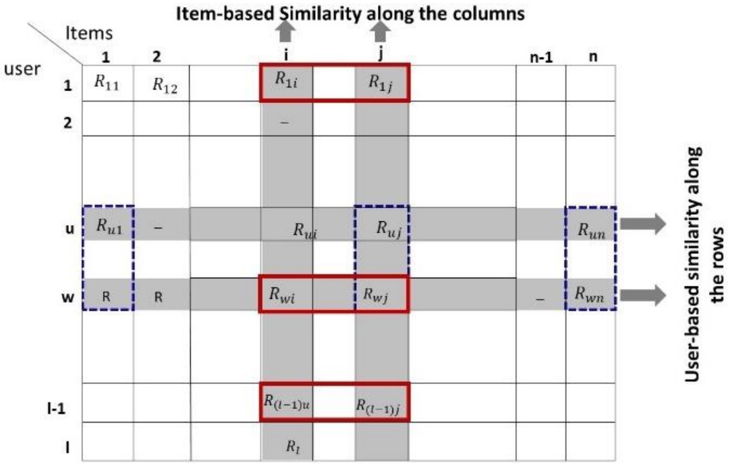


Figure 4. User-item rating matrix

3.4.3. Total item similarity score

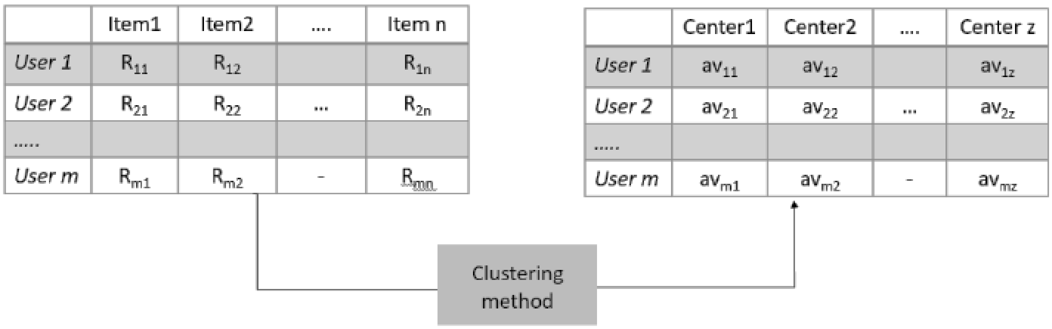
The total similarity between items is obtained by combining the similarity score supplied by ontology in (2) and (3), and explicit user ratings in (4).

$$\text{Similarity } (It_i, It_2) = \alpha\, Sim_{\text{ontology}}\left(It_i, It_j\right) + \mu \text{ Sim}\left(It_i, It_j\right) \quad (5)$$

where $\alpha + \mu = 1$, the total item similarity matrix (TISM) is generated after calculating the overall similarity for each item in the item set using (5).

3.4.4. Method of item clustering

Fuzzy c means clustering [26] was employed in this study to group similar items since it works well with sparse datasets in the majority of recommendation system. This study considers content-based characteristics derived from ontologies combined with user rating data to avoid over generalization, poor accuracy, and cluster overlapping that will result from using just one. As detailed in the next section, similar items within a cluster are used to predict the target item's score. As a result, the number of items that need to be evaluated is significantly fewer than the entire number of items in the system, which increases the system's performance [22]. Once the clusters have been constructed, a user-item cluster matrix (UICM) is produced, in which U represents the set of users, C represents the centers of all item's clusters, and a_{mz} represents the value of the average rating supplied by user m to the item of the cluster center z, as illustrated in Figure 5.



	Item1	Item2	....	Item n
User 1	R_{11}	R_{12}		R_{1n}
User 2	R_{21}	R_{22}	...	R_{2n}
.....				
User m	R_{m1}	R_{m2}	-	R_{mn}

	Center1	Center2	....	Center z
User 1	av_{11}	av_{12}		av_{1z}
User 2	av_{21}	av_{22}	...	av_{2z}
.....				
User m	av_{m1}	av_{m2}	-	av_{mz}

Figure 5. Construction of UICM from user-item rating matrix

3.4.5. Prediction for the rating

Based on the cluster generated, a sorted list of top T similar items is produced for a target item. Using the obtained values, the empty cells in the user-item rating matrix for the target user are then filled. The rating for each unrated (target) item is anticipated based on the active user's ratings for items comparable to that unrated (target) item. Based on (6), we can predict what the rating of an unrated item i will be expected from a target user u.

$$Pred_{u,i} = \frac{\sum_{j\varepsilon T} \text{ Similarity } (i,j)\, rat_{u,j}}{\sum_{j\varepsilon T} \text{ Similarity } (i,j)} \quad (6)$$

Where the similarity score between target item i and item j is *Similarity (i, j)*; $rat_{u,t}$ is the rating for similar item j by user u, and T is the total number of similar items considered. In certain cases, the current user may not rate the top T similar items for a target item, leaving some empty cells in the user-item matrix after filling it. To address this issue, an extended technique might be used to estimate the remaining sparse cells. In this method, an active user's rating behavior for other items is taken into account, as well as other users' ratings for the unrated (target) item. Using the suggested method, an unrated item I may be predicted by target user u as (7):

$$Est_{u,i} = \alpha \frac{\sum_{m=1}^{m} r_{u,m}}{M} + \mu \frac{\sum_{p=1,p\neq u}^{n} r_{p,i}}{n} \quad (7)$$

Where α and μ are control parameters, M is a measure of how many other items U (target user) scored, 1≤m≤M, $r_{u,m}$ is the rating provided by u to other items M, n is the number of other users, where 1≤p≤n, q≠u is the number of other users who submitted a rating for unrated target item I, and $r_{p,i}$ is the rating given to target item i by other users p, excluding target user u. The result for predicting the unrated value in an explicit user-item rating matrix *UIM(U, I)* is a dense, non-sparse user-item rating matrix *DUIM(U, I)*.

3.5. Dimensionality reduction

3.5.1. Singular value decomposition

According to Zhou *et al.* [19], one of the standard solutions for sparsity issues is to use data dimension reduction techniques, notably SVD, which is a matrix factorization technique that can extract dataset features by dividing the original user-item rating matrix into three smaller matrix multiplications. Given a mxn matrix A ∈ R^{mxn} (n is the number of items and m is the number of users), the SVD(A) is expressed with the rank(A)=r as: $SVD(A) = U \times \Sigma \times V^T$, where U ∈ R^{mxr}, V ∈ R^{rxr}, and Σ ∈ R^{rxn}. The middle matrix Σ is a diagonal matrix with r nonzero entries, which are the singular values of A. SVD is the best low-rank linear approximation of the original matrix, which provide the optimum approximation of the utility matrix A.

3.5.2. Incremental singular value decomposition algorithm in the prediction task

The algorithms in the proposed study operate in two stages, online and offline. In the suggested CF recommendation system, user-to-user mapping takes place offline. In contrast, the actual rating prediction or target user interactions is made online. Offline prediction or recommendation is, in fact, a time-consuming procedure. Whereas the online method is efficient in terms of prediction and recommendation time owing to the usage of the incremental SVD. The parallel design system for the similarity formation method may be made incredibly scalable using SVD size reduction techniques while generating more significant results in maximum instances. This study presents incremental SVD algorithms that produce recommendations online for target users in the shortest time possible. The incremental algorithm's most essential quality is that it supports a high number of users, making the system scalable as the size of the user-item matrix grows.

Our recommender system operates in two distinct phases. First, the model is developed offline by calculating user-user or item-item similarity. Meanwhile, the model generates predictions when a newcomer or item is introduced, and the online process begins. In incremental SVD, the projection method is known as folding-in. To fold new users into the distance of the previously decreased user-item matrix. For instance, Figure 6 shows that after running the SVD method on A1 in the offline process with three matrices U1, Σ1, and V1, the online process uses the incremental approach whenever a new matrix A2 is added, resulting in three updated matrices U2, Σ2, and V2.

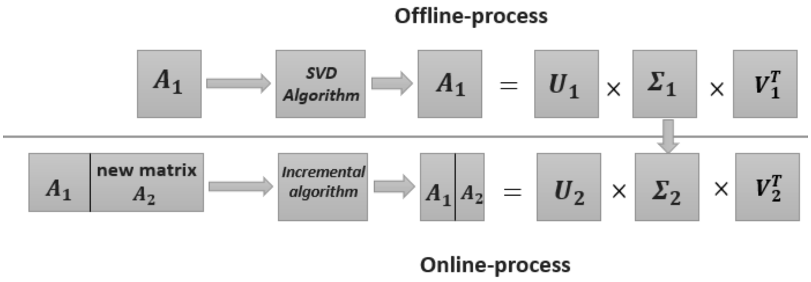


Figure 6. Phases of recommendation process

4. EXPERIMENTATIONS AND RESULTS

4.1. Dataset description

MovieLens dataset, which can be found at [27], is one of the most well-known datasets for evaluating recommender systems. The MovieLens dataset consists of 1 M ratings provided by 6,040 users for a total of 3,900 movies. Each rating is expressed on a scale of 1 to 5, where a rating of 1 indicates the least liked movie and a rating of 5 represents the most liked movie. The dataset offers a comprehensive collection of user reviews, allowing us to evaluate and enhance our recommendation system based on a broad range of user preferences and movie ratings. Detailed information about the dataset is presented in the Table 1.

Table 1. Description of the dataset

	MovieLens 1 M
User#	6,040
Items# (movies)	3,900
Ratings#	1,000,209
Rang of ratings	1-5
Sparsity#	95.75%

WebSPHINX [28], a web crawler, was used in this study to collect material relevant to IMDb [29] items. Furthermore, gathered data is used to construct and complete an item ontology. To conduct tests, the dataset was divided into 80% of randomly selected data for the training set, while 20% of the remaining data was used for the testing set.

4.2. Evaluation and discussion of the proposed system

The recommender system presented in this study was implemented using Python 3.9.7 on a PC with a 4 GHz processor, 8 GB RAM, and 64-bit Microsoft Windows 10. To thoroughly assess the system's performance, it was compared to various related approaches, including Pearson nearest neighbor algorithm, item-based CF with EM, SVD combined with ontology, and user-item-based EM and SVD with and without ontology integration. The evaluation was conducted from two perspectives: time throughput (recommendations per second) and accuracy, providing valuable insights into the system's efficiency and effectiveness compared to existing approaches.

4.2.1. Evaluation 1: predictive accuracy analysis

Mean absolute error (MAE) is a statistical accuracy metric used to evaluate prediction accuracy. In this experiment, the MAE computes the difference between the predicted and actual ratings. MAE is presented in (8):

$$\text{MAE}(\text{pred}, act) = \sum_{k=1}^{N} \left|\frac{\text{pred}_{u,k} - act_{u,k}}{Nb}\right| \tag{8}$$

Where Nb determines the number of items on which a user u has given a score, the suggested approach for predicting accuracy using MAE is assessed and compared to the state-of-the-art methods. Displayed on MovieLens datasets, respectively, against different neighborhood sizes Figure 7.

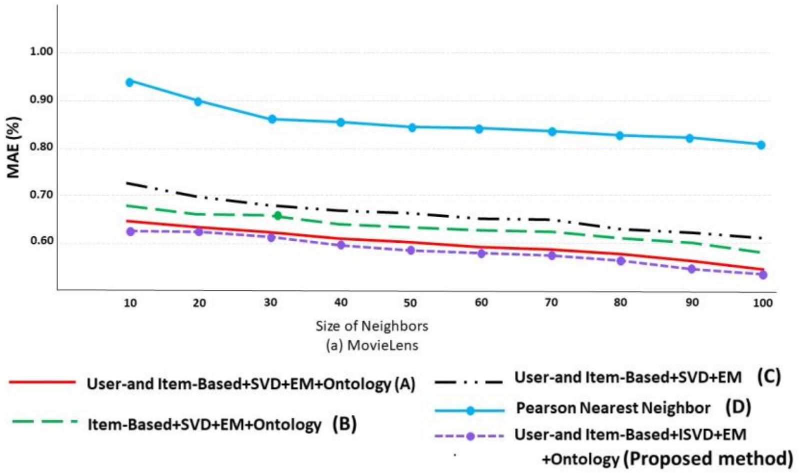


Figure 7. MAE using MovieLens datasets for all methods

4.2.2. Evaluation 2: decision-support accuracy

In terms of accuracy measurements, the decision-support metrics will be crucial in evaluating the overall performance of the hybrid-based recommender. In the information retrieval area, several measures for this aim are well-known. Recall, precision, and F-measure are among the metrics included in this category. The precision computes the fraction of relevant items in the list of returned results. In contrast, the recall calculates the fraction of pertinent items that have been retrieved. Both metrics should be used in common since the recall increases as the number of items retrieved increases, whereas the precision often decreases as result sizes increase.

$$\text{precision } = \frac{TR}{TR+FR} \tag{9}$$

$$\text{recall } = \frac{TR}{TR+FN} \tag{10}$$

The F measure is a metric that takes both values into account, as indicated in (11):

$$F1 = \frac{(1+\alpha^2)\cdot \text{precision} \cdot \text{recall}}{\alpha^2\cdot \text{precision} + \text{recall}} \tag{11}$$

The F1 measures and precision values for all methods on various top-N recommendations are shown in Table 2. It can be deduced from the table that the precision achieved by the suggested technique is significantly higher than that obtained by the nearest neighbor algorithm or the other methods tested. In addition, we found that the F1 measures of the proposed method, dealing with dimensionality reduction using incremental SVD and ontology, outperformed. Compared to other methods, these findings are sufficient to support our claim that our recommendation system is reasonably more efficient and scalable.

Table 2. Comparison of F1 metric and the precision values for different methods

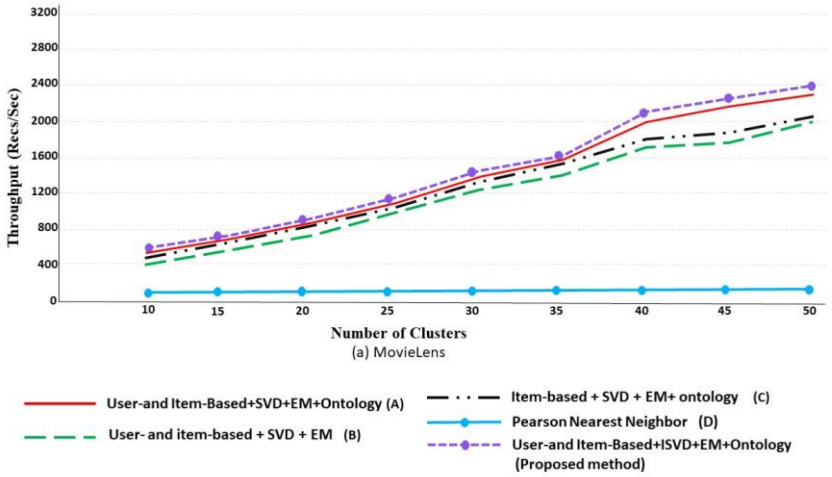
Top N	Proposed system		Method A		Method B		Method C		Method D	
	F1	Precision	F1	Precision	F1	Precision	F1	Precision	F1	Precision
Top-5	0.811	0.801	0.797	0.787	0.773	0.771	0.721	0.719	0.583	0.564
Top-10	0.816	0.805	0.807	0.796	0.784	0.782	0.739	0.736	0.601	0.582
Top-15	0.83	0.82	0.827	0.816	0.804	0.802	0.749	0.747	0.615	0.592
Top-20	0.843	0.832	0.833	0.821	0.811	0.809	0.76	0.757	0.622	0.601
Top-25	0.856	0.845	0.844	0.833	0.825	0.823	0.77	0.769	0.65	0.628
Top-30	0.845	0.837	0.84	0.831	0.821	0.819	0.762	0.757	0.605	0.603
Top-35	0.837	0.83	0.832	0.823	0.808	0.806	0.751	0.75	0.59	0.581
Top-40	0.839	0.828	0.83	0.819	0.803	0.801	0.741	0.739	0.579	0.573
Top-45	0.833	0.819	0.827	0.818	0.795	0.793	0.732	0.733	0.558	0.556
Top-50	0.831	0.817	0.822	0.813	0.785	0.783	0.722	0.723	0.546	0.541

Method A: user-and item-CF based+SVD+EM+ontology
Method B: item-CF based+SVD+EM+ontology
Method C: user-and item-CF based+SVD+EM
Method D: pearson nearest neighbor

4.2.3. Evaluation 3: scalability analysis

The efficiency of the suggested approach is evaluated in the first experiment. Evaluation is based on throughput, known as the number of suggestions per second. We test our strategy on the MovieLens datasets to demonstrate its effectiveness in improving the system's scalability problem. Figure 8 illustrates the performance results of our method compared to the state-of-the-art methods.

According to the graph, the throughput of those methods that use dimensionality reduction techniques and clustering is considerably higher than other methods. Moreover, the proposed approach based on clustering with expectation maximization (EM), ontology similarity, and incremental SVD is slightly higher than other methods, especially those that rely on the SVD reduction technique. Unlike systems that use the nearest neighbor technique, clustering allows the recommendation system to analyze just a part of the items/users. As a result, increasing the cluster size does not affect throughput since it must scan all nearest neighbors.



(a) MovieLens

Figure 8. Throughput of all methods for MovieLens datasets

The results of the evaluations demonstrate the effectiveness and superiority of the proposed recommendation system. By incorporating ontology and dimensionality reduction techniques in CF, the system achieves improved predictive accuracy, decision-support accuracy, and scalability compared to existing methods. It proved that considering semantic relationships and reducing dimensionality enhances the system's ability to capture user preferences, provide accurate ratings, and enable the system to handle large-scale datasets more effectively. Therefore, this implies that the proposed method not only provides accurate recommendations but also ensures that relevant items are retrieved. The system addresses the limitations of traditional CF approaches by providing accurate recommendations, assisting users in decision-making, and efficiently handling large datasets. This study's findings highlight the proposed system's potential for practical applications in the recommendation domain.

5. CONCLUSION

In this paper, we have presented a novel recommendation method that addresses the challenges of accuracy, scalability, and sparsity in CF-based recommender systems. Our approach incorporates dimensionality reduction using the incremental SVD algorithm, ontological item-based semantic similarity, and explicit user ratings to improve the prediction accuracy and scalability of the system. By adopting the incremental SVD method, we were able to handle the increasing size of the user-item matrix while maintaining computational efficiency. The folding-in technique employed in the incremental SVD algorithm significantly reduced the computation cost and allowed our system to achieve high scalability. The experimental results conducted on a real-world movie recommendation dataset confirmed the effectiveness of our proposed method. The precision, F1 measures, and MAE metrics demonstrated that our system provides accurate predictions while effectively addressing the sparsity and scalability issues commonly encountered in recommender systems. The incorporation of MO further improved the predictive accuracy and expanded the potential for applying our method to different semantic contexts and domains. Further research can explore additional evaluation metrics, investigate the system's performance with different datasets, and consider the impact of incorporating other factors such as user demographics or temporal dynamics. By continuously refining and enhancing the recommendation system, we can further improve the accuracy, relevance, and usability of the recommendations provided to users in various domains.

REFERENCES

[1] Z. Fayyaz, M. Ebrahimian, D. Nawara, A. Ibrahim, and R. Kashef, "Recommendation systems: Algorithms, challenges, metrics, and business opportunities," *Applied Sciences*, vol. 10, no. 21, pp. 1–20, 2020, doi: 10.3390/app10217748.

[2] S. K. Addagarla and A. Amalanathan, "A survey on comprehensive trends in recommendation systems & applications," *International Journal of Electronic Commerce Studies*, vol. 10, no. 1, pp. 65–88, 2019, doi: 10.7903/ijecs.1705.

[3] H. Jafarkarimi, A. T. H. Sim, and R. Saadatdoost, "A naive recommendation model for large databases," *International Journal of Information and Education Technology*, vol. 2, no. 3, pp. 216–219, 2012.

[4] H. Khojamli and J. Razmara, "Survey of similarity functions on neighborhood-based collaborative filtering," *Expert Systems with Applications*, vol. 185, pp. 1–28, 2021, doi: 10.1016/j.eswa.2021.115482.

[5] G. Adomavicius, N. Manouselis, and Y. Kwon, "Multi-Criteria Recommender Systems," in *Recommender Systems Handbook*, Boston, MA: Springer, 2011, pp. 769–803, doi: 10.1007/978-0-387-85820-3_24.

[6] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005, doi: 10.1109/TKDE.2005.99.

[7] R. Katarya and O. P. Verma, "An effective collaborative movie recommender system with cuckoo search," *Egyptian Informatics Journal*, vol. 18, no. 2, pp. 105–112, 2017, doi: 10.1016/j.eij.2016.10.002.

[8] H. El Massari, N. Gherabi, S. Mhammedi, H. Ghandi, F. Qanouni, and M. Bahaj, "Integration of ontology with machine learning to predict the presence of covid-19 based on symptoms," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 5, pp. 2805-2816, Oct. 2022, doi: 10.11591/eei.v11i5.4392.

[9] S. Mhammedi, H. El Massari, and N. Gherabi, "Composition of Large Modular Ontologies Based on Structure," in *Lecture Notes in Networks and Systems*, Cham: Springer, 2022, pp. 144–154, doi: 10.1007/978-3-030-91738-8_14.

[10] H. El Massari, N. Gherabi, S. Mhammedi, H. Ghandi, F. Qanouni, and M. Bahaj, "An Ontological Model based on Machine Learning for Predicting Breast Cancer," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 7, pp. 108-115, 2022, doi: 10.14569/IJACSA.2022.0130715.

[11] C. M. -Cruz, C. Porcel, J. B. -Moreno, and E. H. -Viedma, "A model to represent users trust in recommender systems using ontologies and fuzzy linguistic modeling," *Information Sciences*, vol. 311, pp. 102–118, 2015, doi: 10.1016/j.ins.2015.03.013.

[12] M. A. -Hassan, H. Lu, and J. Lu, "A semantic enhanced hybrid recommendation approach: A case study of e-Government tourism service recommendation system," *Decision Support Systems*, vol. 72, pp. 97–109, 2015, doi: 10.1016/j.dss.2015.02.001.

[13] N. R. Kermany and S. H. Alizadeh, "A hybrid multi-criteria recommender system using ontology and neuro-fuzzy techniques," *Electronic Commerce Research and Applications*, vol. 21, pp. 50–64, 2017, doi: 10.1016/j.elerap.2016.12.005.

[14] S. Ayesha, M. K. Hanif, and R. Talib, "Overview and comparative study of dimensionality reduction techniques for high dimensional data," *Information Fusion*, vol. 59, pp. 44–58, 2020, doi: 10.1016/j.inffus.2020.01.005.

[15] S. Mhammedi, H. El Massari, N. Gherabi, and M. Amnai, "CF Recommender System Based on Ontology and Nonnegative Matrix Factorization (NMF)," in *Artificial Intelligence and Smart Environment*, Cham: Springer, 2023, pp. 313–318, doi: 10.1007/978-3-031-26254-8_44.

[16] N. T. Son, D. H. Dat, N. Q. Trung, and B. N. Anh, “Combination of Dimensionality Reduction and User Clustering for Collaborative-Filtering,” in *ACM International Conference Proceeding Series*, pp. 125–130, 2017, doi: 10.1145/3168390.3168405.

[17] J. Wang, P. Han, Y. Miao, and F. Zhang, “A Collaborative Filtering Algorithm Based on SVD and Trust Factor,” in *International Conference on Computer, Network, Communication and Information Systems (CNCI 2019)*, 2019, pp. 33–39, doi: 10.2991/cnci-19.2019.5.

[18] M. Nilashi, O. Ibrahim, and K. Bagherifard, “A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques,” *Expert Systems with Applications*, vol. 92, pp. 507–520, 2018, doi: 10.1016/j.eswa.2017.09.058.

[19] X. Zhou, J. He, G. Huang, and Y. Zhang, “SVD-based incremental approaches for recommender systems,” *Journal of Computer and System Sciences*, vol. 81, no. 4, pp. 717–733, 2015, doi: 10.1016/j.jcss.2014.11.016.

[20] M. Brand, “Incremental singular value decomposition of uncertain data with missing values,” in *Computer Vision—ECCV 2002*, Berlin, Heidelberg: Springer, 2002, pp. 707–720, doi: 10.1007/3-540-47969-4_47.

[21] A. Bouza, “Movie Ontology,” *GitHub*, 2014. [Online]. Available: https://github.com/amanciobouza/movieontology/blob/e87f858432af3905ff69179fa5aa859f9e03563c/movieontology.owl. Accesed: May 15, 2022.

[22] N. Bassiliades, M. Symeonidis, G. Meditskos, E. Kontopoulos, P. Gouvas, and I. Vlahavas, “A semantic recommendation algorithm for the PaaSport platform-as-a-service marketplace,” *Expert Systems with Applications*, vol. 67, pp. 203–227, 2017, doi: 10.1016/j.eswa.2016.09.032.

[23] M. I. M. -Vicente, A. G. -Solla, M. R. -Cabrer, Y. B. -Fernández, and M. L. -Nores, “Semantic inference of user’s reputation and expertise to improve collaborative recommendations,” *Expert Systems with Applications*, vol. 39, no. 9, pp. 8248–8258, 2012, doi: 10.1016/j.eswa.2012.01.162.

[24] N. H. Pribadi, R. Sarno, A. S. Ahmadiyah, and K. R. Sungkono, “Semantic Recommender System Based on Semantic Similarity Using FastText and Word Mover’s Distance,” *International Journal of Intelligent Engineering and Systems*, vol. 14, no. 2, pp. 377–385, 2021, doi: 10.22266/ijies2021.0430.34.

[25] A. Daoui, N. Gherabi, and A. Marzouk, “A new approach for measuring semantic similarity of ontology concepts using dynamic programming,” *Journal of Theoretical and Applied Information Technology*, vol. 95, no. 17, pp. 4132–4139, 2017.

[26] R. Katarya and O. P. Verma, “An effective web page recommender system with fuzzy c-mean clustering,” *Multimedia Tools and Applications*, vol. 76, no. 20, pp. 21481–21496, 2017, doi: 10.1007/s11042-016-4078-7.

[27] “MovieLens: non-commercial, personalized movie recommendations,” *MovieLens*. [Online]. Available: https://movielens.org/. Access date: Jul. 24, 2023.

[28] “WebSPHINX: A Personal, Customizable Web Crawler,” *WebSPHINX*. [Online]. Available: https://www.cs.cmu.edu/~rcm/websphinx/. Access date: Jul. 24, 2023.

[29] “IMDb: Ratings, Reviews, and Where to Watch the Best Movies & TV Shows,” *IMDb*. [Online]. Available: https://www.imdb.com/. Access date: Jul. 24, 2023.

BIOGRAPHIES OF AUTHORS

Sajida Mhammedi received her master's degree in business intelligence (BI) from the Faculty of Science and Technology of Beni Mellal, Morocco. Currently, she is a researcher at the Laboratory of Science and Technology for Engineering (LaSTI), at the National School of Applied Sciences, Sultan Moulay Slimane University, Khouribga, Morocco. Her research interests include machine learning, deep learning, big data, intelligent systems, semantic web, and ontology. She can be contacted at email: sajida.mhammedi@usms.ac.ma and sajidamh@gmail.com.

Noreddine Gherabi is a professor of computer science with industrial and academic experience. He holds a doctorate degree in computer science. In 2013, he worked as a professor of computer science at Mohamed Ben Abdellah University and since 2015 has worked as a research professor at Sultan Moulay Slimane University, Morocco. Member of the International Association of Engineers (IAENG). He has several contributions in information systems namely: big data, semantic web, pattern recognition, and intelligent systems. He has papers (book chapters, international journals, and conferences/workshops) and edited books. He has served on executive and technical program committees and as a reviewer of numerous international conferences and journals, he convened and chaired more than 30 conferences and workshops. He is member of the editorial board of several other renowned international journals: i) co-editor in chief (editorial board) in the journal “The International Journal of sports science and engineering for children” (IJSSEC), ii) associate editor in the journal “International Journal of Engineering Research and Sports Science”, iii) reviewer in several journals/conferences, and iv) excellence award, the best innovation in science and technology 2009. His research areas include machine learning, deep learning, big data, semantic web, and ontology. He can be contacted at email: n.gherabi@usms.ma.

Hakim El Massari received his master's degree from Normal Superior School of Abdelmalek Essaadi University, Tétouan, Morocco, in 2014. Currently, he is preparing his Ph.D. in computer science at the National School of Applied Sciences, Sultan Moulay Slimane University, Khouribga, Morocco. His research areas include machine learning, deep learning, big data, semantic web, and ontology. He can be contacted at email: h.elmassari@usms.ma.

Zineb Sabouri received her diploma in Computer Engineering degree from the National School of Applied Sciences of Khouribga, Morocco. She worked as a computer engineer in a multinational. Currently, she is a Ph.D. student in computer science at Sultane Moulay Slimane University. Her area of interest is machine learning, intelligent systems, deep learning, and big data. She can be contacted at email: zineb.sabouri@usms.ac.ma

Mohamed Amnai received his Ph.D. on telecommunications and computer science, from Ibn Tofail University in Kenitra city, Morocco. Since March 2014, he is a professor. He has been professor at National School of Applied Sciences Khouribga, Settat University, Morocco. Now, he is professor in Computer Science Research Laboratory, Ibn Tofail University, Kenitra, Morocco. He can be contacted at mohamed.amnai@uit.ac.ma.