Two recommendation system algorithms used SVD and association rule on implicit and explicit data sets

Article *in* International Journal of Scientific & Technology Research · January 2020

3 authors:

Marwa Hussien Mohamed
October 6 University

Mohamed Hasan Ibrahim
Fayoum University

Mohamed Khafagy
Fayoum University

Two Recommendation System Algorithms Used SVD And Association Rule On Implicit And Explicit Data Sets

Marwa Hussien Mohamed, Mohamed Helmy Khafagy, Mohamed Hasan Ibrahim

Abstract: Nowadays, the recommender system is an important research area for online companies that suggest items and services to users like (last FM music, Netflix movies, and movie-lens). Building a recommendation system to meet users' preferences is very difficult due to rapidly increasing the size or volume of digital information. Also, the recommendation has many challenges that need to overcome like sparsity, accuracy, performance and novelty. In this paper, we build two new algorithms to solve the sparsity, accuracy and performance of the recommendation system. Firstly, we used association rule mining to find a hidden pattern and count numbers of played songs per transaction and compute similarities by cosine vector similarity to make a recommendation to users also taking into concern the rating merged with clustering technique. Secondly, we used K-means clustering algorithms with SVD (singular value decomposition) to reduce dimensionality, increase the performance, and solve sparsity and accuracy problems. Our experiments are applied on last FM music datasets and movie-lens datasets implicit and explicit feedback, we compare our new algorithms with k-means collaborative filtering using RMSE (root mean square error) to show the accuracy and performance of movie lens and measure the accuracy using precision, recall and, F- measure to show the accuracy between basic collaborative filtering and our two new algorithms. This experiment shows that using association rule is better than improved k-means while combining with SVD and basic collaborative filtering. But our new k-means and SVD algorithm has better performance than random collaborative filtering K-means.

Index Terms: Recommender systems; K-means clustering; Association Rule; SVD (Singular Value Decomposition); Dimensionality Reduction.

1. INTRODUCTION

IN this modern era, everybody depends on the internet to find products, services, and items daily to determine their needs. This is regarded as a natural phenomenon of the human decision-making process [1]. Recommender systems help users to get the decision more easily and rapidly by filtering huge information about millions of products and items over websites. The main issue is to recommend items to users' will be liked and give a high rate to meet their expectations. Recommender system process using some data mining techniques like clustering to group similar items together to find similarity between them when we select an item as a centroid of the cluster or user's similarity if we select a user as a centroid of the cluster and association rules to find hidden patterns and discover new relationships between products to increase sales as a part of E-commerce [2].

Recommender system has many techniques [3]:

- Content based filtering: it's depends on analyzing the content of textual data and discovering similarities between items specification.
- Collaborative filtering techniques: it's depends on the similarities between users' rating items on the site to recommend to the other different presences he or she may be liked.
- Hybrid collaborative filtering: it's merging between content based filtering and collaborative filtering to gain more advantage and get best recommended items results [4].

Recommender systems work by using feedback on products as data inputs to the systems. It has two types explicit and implicit data sets. The feedback mostly used is explicit, it calculates the similarity and makes recommendations depends on users' ratings on items. The other type is used several watching items like movies or listening to songs more times or viewing product types this named by implicit feedback. But, one of recommender systems challenges is users' feedback about items it's not available all times this called sparsity problems so that recommender systems can use implicit feedback which it reflects users' preferences indirectly to the system [5]. For recommendation system example, if users like to watch a lot of movies for an actor probably this user likes this actor so we can recommend to him different movies haven't seen on the website before [5]. Figure 1 shows an example for reading and rating books between three users and four books the first and the third user when they rate one of the four books its explicit feedback, but the second user wish and reading some of these books without rating it's implicit feedback.

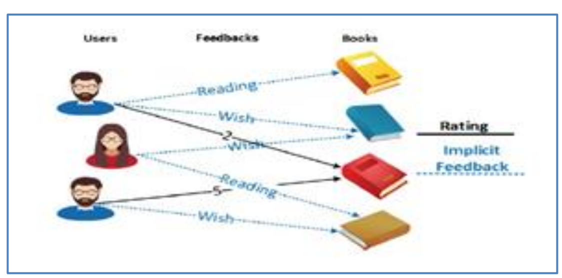


Figure 1. *Example of implicit and explicit feedback.*

Another example of explicit feedback like movies-lens

- *Marwa hussien Mohamed, Department of Information systems, October 6 University, Fayoum University, Cairo, Egypt, E-mail: Eng_maroo1@yahoo.com*
- *Mohamed Helmy Khafagy, Department of Computer Science, Fayoum University, Cairo, Egypt, E-mail: Mhk00@fayoum.edu.eg*
- *Mohamed Hasan Ibrahim, Department of Information systems, Fayoum University, Cairo, Egypt, E-mail: mhi11@fayoum.edu.eg*

using users rating about movies to recommend items to other users like if the users give 5 stars it's much liked and if the user gives 1 star, he is extremely disliked. Another example of Implicit feedback like how many time users listen to this song without rating this song we can take it into concern this number to make another recommendation to this user with the same type of this song. This paper solves sparsity problem and increase accuracy of recommendations by using two techniques one of them using association rules with clustering is a type of data mining techniques to find hidden patterns and count number of played songs as an implicit feedback added to collaborative filtering techniques with real rating from users and another one using based k-means clustering algorithm combined with SVD (singular value decomposition) to reduce dimensionality which it's used to improve scalability of recommender systems. Our experiments applied on implicit and explicit feedback of last.FM music datasets and Movie lens datasets. The rest of this paper is organized as follows: a recommender system in section 2. The Proposed algorithms are discussed in section 3. The Experimental evaluation discussed in section 4. Finally, a conclusion and future work is given in Section 5.

2 RECOMMENDER SYSTEMS

Recommender systems are good for online shopping environments that are reducing the cost of searching transaction and increase buying items at a reasonable time. It's good for users' to find their interests rabidly and increase the number of sales for products and items online. Also, the recommender system can improve the decision-making process [6]. Recommender system [9] used to filter out information to users between a lot of daily needs and millions of services and products like selecting their preferences from movies, music, news, images, web pages. If we have a fresh user to the system[3], it's named by cold start challenges we need to know what he need or like from our site so we can use demographic filtering to recommend items suitable for his living place educational level and we can make a survey to the user and recommend some of the items. If he liked this, we can recommend other products has the same specification it's content-based filtering or recommend items selected by other users like the same item it's collaborative filtering. Figure 2 shows recommender system component [9].

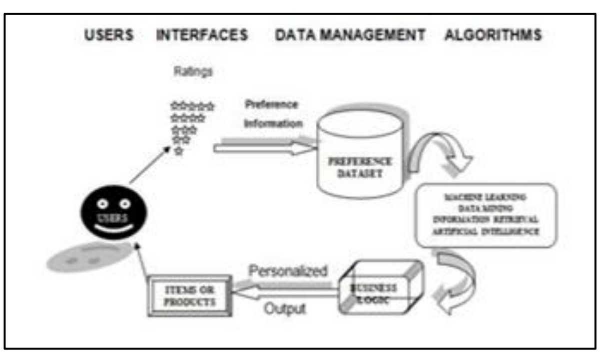


***Figure 2**. Recommender system components*

Recommendations can be presented to an active user in two different ways [1]:

1. Predicting ratings of items that the user has not seen.
2. Construct a list of items ordered by the users' preferences, which is known as top-N recommendations.

2.1 Collaborative Filtering

Collaborative filtering is a preferred technology in a recommendation system. It has three features are a) Popularity, b) User profile, c) Activity to help us connect between users or any items and topic [8]. The fundamental assumption of CF is that if users X and Y rate n items similarly, or have similar behaviors (e.g., buying, watching, listening), and hence, they may give rate or act on other products similarly [3].

Collaborative filtering has two types: Model-based and memory-based table 1 list advantages and disadvantage of them [3].

TABLE 1

OVERVIEW OF COLLABORATIVE FILTERING TECHNIQUES

CF categories	*Main advantages*	*Main shortcomings*
Memory-based CF	*∗easy implementation ∗new data can be added easily and Incrementally. ∗need not consider the content of the items being recommended. ∗scale well with correlated items.*	*∗Dependent on human ratings. ∗performance decrease when data are sparse ∗cannot recommend for new users and new items ∗limited scalability for large datasets.*
Model-based CF	**better address the sparsity, scalability and other problems. *improve prediction performance. ∗give an intuitive rationale for the recommendations.*	*∗ expensive model-building ∗ have a trade-off between prediction performance and scalability ∗lose useful information for dimensionality reduction techniques*

The most famous similarity measures are Cosine Vector Similarity [9] (CVS) equation 1.

$$W_{Xy} = \frac{x \cdot y}{|x| \times |y|} = \frac{\sum_{i=1}^{n} (r_{Xi})(r_{yi})}{\sqrt{\sum_{i=1}^{n} (r_{Xi})^2}\sqrt{\sum_{i=1}^{n} (r_{yi})^2}} \tag{1}$$

Where x and y are users rated number of items n. the average rating of item x is rx and an average rating of item y are ry. The weight measures between the preferences /interests of users are WXy ∈ [-1, 1]. Also, rxi and ryi are the ratings of the users x and y on item i. the similarity between active users are measured by any of these two equations. We can use eq (2) to predict the weighted average of rating on items like j by the formula predXj by using all neighbors of active users. The set of neighbors is represented by K [9].

$$pred_{Xj=}\overline{r_x} + \frac{\sum_{y=1}^{k} \left(r_{yj} - \overline{r_y}\right) \times w_{xy}}{\sum_{y=1}^{k} |W_{Xy}|} \tag{2}$$

Although, these equations used to compute the similarity are successful to find the nearest neighbor users but have some difficulty when data are spare because many users don't like to rate purchased items on the system [9].

2.2 Clustering in recommender systems
Clustering is used with unlabeled dataset to find patterns like Machine learning, image analysis, pattern recognition and outlier detection are few of many application areas of clustering. It's one of recognizing patterns and unsupervised classification used to group objects near similar together [1].
Type of clustering [7]

1. Centroid-Based Clustering
2. Distributed-Based Clustering
3. Connectivity-Based Clustering
4. Density-Based Clustering

2.3 K-means Clustering
K-means Clustering [7] is widely used for clustering algorithms todays because of his ability to handle large volume of data sets in a reasonable time to increase the performance. This algorithm partitions the whole datasets in a number of disjoint clusters named by K. K-means general algorithm steps should start by defining the number of K clusters. Then we select randomly some of our data sets to be taken as a cluster's center. The used the Euclidean distance matrix to scan the remaining datasets and send these data to the closest cluster [10]. Every time we need to calculate the mean of the cluster to update the mean value. This process is repeated with new centroid values and all points reassigned to the new clusters. The Centroid's value is changed after iteration. This will be the end when the cluster centers value has no change.

2.4 Recommender system challenges
- Sparsity: collaborative filtering suffers from sparse data while comparing items using memory-based algorithms and it's difficult to get accurate predictions [3].
- Cluster quality: best recommendation come from but items similarly in the same cluster to recommend this new item to the user clustering data is a very complicated task [1].
- Accuracy: it's an important challenge of the system because if we recommend wrong items to the user, he may leave the site or system. So that we need to recommend items has high accuracy results [11].
- Scalability: the system will be able to handle the recommended items while increasing the users and items number in a better time. One of the problems of collaborative filtering algorithms suffers from scalability issues [12].

3 PROPOSED ALGORITHMS
In this paper, we build two new algorithms to solve sparsity and accuracy challenge of recommendation systems. Collaborative filtering techniques suffer from small rating for data items on website to recommend new items to users. We will use implicit feedback data and explicit to measure the efficiency of our algorithms to solve these problems.

3.1 Problem definition
Clustering algorithm K-means choose randomly the initial centroids of the cluster. This may lead to poor quality clustering results. We need to solve this problem to get a more accurate recommendation for the traditional k-means with collaborative filtering and SVD singular value decomposition in our new algorithm. We should maximize the coverage of the recommendation algorithm. Recommendation algorithms should work well and output good recommendation to users while increasing the data size to solve sparsity and accuracy challenges. Accuracy challenges will be solved by merging between Association rule mining and clustering techniques. Association rule mining used efficiency with massive datasets because it's previously used to capture a total purchase made (the number of times a user has purchased or used a product), now we will count the number of total purchases per transaction with implicit and explicit feedback data. Association rule mining will model user's behavior on the system and to find the using of repeated items with transactions.

3.2 New K-means and SVD proposed algorithm
This proposed algorithm we need to build an effective recommendation system to output recommended models has a great accuracy by combining the characteristics of K-means clustering based for divide the data into a number of clusters contain users and its data by selecting the centroid k of the clusters randomly and reduce the size of data to find similarities between users fast. Then we will use the capability of SVD technique to reduce dimensionality by reducing number of unused features of data. This algorithm has two steps, one of them was built offline named by offline model creation and the second done online while we recommend and produce accurate recommendations to users active. To measure similarity between users we use cosine vector similarity.

K-means clustering steps:

1. We will use K as the number of clusters and user-item rating matrix as input
2. Selecting the initial k users clustering centers randomly
3. We will assign users to the nearest cluster by measuring the distance between k users' center and other users.
4. We need to repeat the calculation of new partition centers for each user's cluster
5. If there is a change in partition center, we need to redistribute users again to the clusters
6. Repeats the last two steps again till we have no change in centers
7. The output center- items rating matrix is a K clusters.

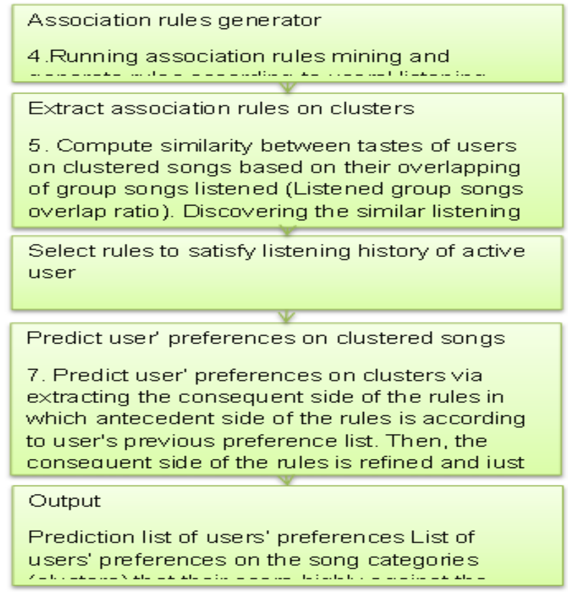
The first step for the algorithm is an offline model it gets the input value to the system item matrix include a rating value to the songs as explicit feedback and the number of playing songs as implicit feedback to solve sparsity challenges. Create user clusters using k-means steps as we discuss previously and stop when we find no change in the mean of the clusters and algorithm converge are to a stable partition. Then apply SVD to find the decomposition matrices after that we calculate the similarity between users and output a recommendation model. We do a lot of training experiments offline on the recommendations model after that we do prediction and recommendation to the active user online. At the online stage, we use SVD to find the neighbors of active users based on the clusters we divided before using k-means offline. The second step it's an online model to recommend items to active users. We get active user and items and recommendation model as input data to this step. We will use the original matrix that's contained rating values to items and the number of played and listening to the songs also it contains users' clusters, we will find the nearest user in the clusters has the highest similarity and recommend these items to the new user this step output different item we predict using prediction formula to the users near his preferences and previous rated items or listing on the site.

3.3 New Association rules and clustering based proposed algorithm

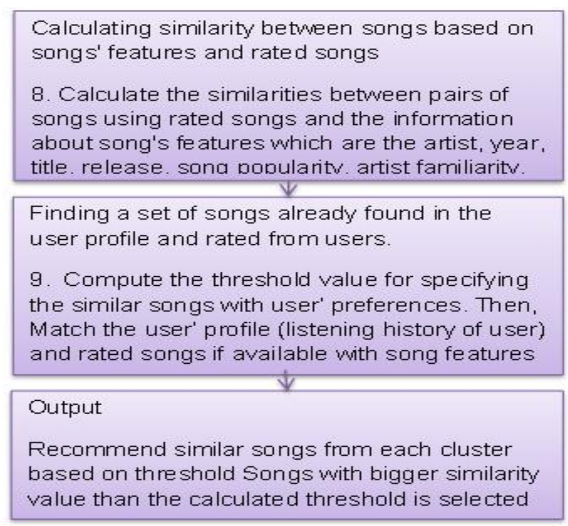
Our new proposed technique used to increase the accuracy of predicted items will be recommended to users also solve the sparsity problem by using the merging between implicit and explicit data. Our data sets are part of million song data sets is last.FM datasets [12] used for the research paper [13]. Our new techniques are best compared to the basic collaborative filtering techniques and new k-means and SVD algorithm. The main advantage of our proposed algorithm is how to find the correlation between items in a spare data this be solved by using implicit data (Songs played, number of play counts to a specific song, play ratio for certain category of songs and tagging information). Also, a similar behavior pattern can compute easily with our new algorithms this led to a good recommendation. The main idea we solve in this technique where data are spare in this case users don't have any co-played music and we need to find similar preferences on pop music. However, such a result is not true in more sparse rating data. Even though both users do not have any co-played music, both are fans of pop music. Thus, we should consider them to be similar in the case that they are sharing a very similar preference for pop music. We will compute the similarity between users on playing songs groups by applying the clustering techniques and association rules to find hidden items between active users, or users interested in items available on the system. figure 3 show new recommender system steps. We will discuss proposed algorithms

Phase 1: *Pre-processing Reducing data dimensionality for the rule mining part*

Phase 2: *Prediction process on users' unknown preferences based on association rule mining*



Phase III: *Recommendation Recommend a list of songs available in each cluster that match the user' preferences*



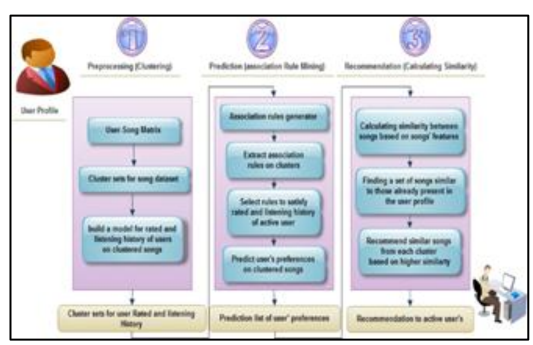


Figure 3.*new proposed algorithm steps*

4 EXPERIMENTAL EVALUATIONS

This section will show the results of our experiments on last.Fm million song data sets with merge between implicit and explicit data to improve the accuracy of recommendation systems [13,14] and using movie-lens data sets[15].We will compare our two new algorithms with traditional collaborative filtering techniques while a change in the data sparsity level. This paper will focus in the music recommendation and movie recommendation so that we use million song datasets [13] and movie lens datasets [15]. Last.Fm it's large and free datasets in the music domain. It is constructed from about one million songs and users, in which each user plays a small set of songs. It's contained implicit feedback for users' preferences and has item matrix spare and Last.Fm dataset for tagging activity of songs. Also, we are using movie-lens 1M data sets rating from 1 to 5 to measure the performance of our algorithms. It has one million of ratings made by six thousand participants rating about four thousand online movies.

4.1 Experimental methodology and evaluation metrics

We use precision and recall equation to measure the accuracy and evaluate these data sets between our new algorithms, and basic collaborative filtering [16]. The description for accuracy metric seen in table 2.

TABLE 2

RECOMMENDATION ACCURACY METRIC

Predicted items/actual	Relevant	Irrelevant
Recommended	True Positive (TP)	False Positive (FP)
Not recommended	False Negative (FN)	True Negative (TN)

The precision measure [16] the ability of the system to return relevant items among a set of irrelevant and relevant items and it's calculated by the equation (3)

Precision=TP/(TP+ FP) (3)

The Recall measure [16] the ability of the system to return the relevant items only and it's calculated by the equation (4).

Recall=TP/(TP+ FN) (4)

Another evaluation metric is an F-measure [16] used to find the difference between precision and recall function and to an equal weight of each of them. The metric equation is (equation 5). The higher result means higher accuracy of recommendation.

F-measure = (2 * Precision * Recall) / (Precision + Recall) (5).

We focus on number rated songs (this an explicit data) on user profile from 1-5 this will indicate the users, which rate 5 to a song he is very likely it and rate from 2-1 he dislikes this song. Also, we focus on user profile preferences implicit data. We divide the data into categories from 1-5 depending on the high number of played songs or listing songs in such way the song In a range 80-100% this like the rating 5, and songs from 1-20 % this like rating 1.

Then we build a user-item rating matrix we will compare with basic collaborative filtering. That's the firstly preprocessing step. In order to find similarities and train models based on user profile preferences like (user-song-play count) 80% as a training dataset and 20 % as a test dataset. These steps applied to last.FM million song datasets and movie-lens datasets. We cluster data sets into three clusters determine the songs level like (level 0 – all songs), (level 1- songs tags (pop, rock, jazz, etc.)) and (level 3 – song duration (very short- one minute or less, short – from 1 minute to 3, medium is form 3 minutes to 5 , long- from 5 minutes to 8, very long- more than 8 minutes)).this used to help association rule mining by defining optimal numbers of clusters and used the songs durations to build it. Also, we need to reduce the size data send to the association rule to get the best performance extraction of an association rule. After this, firstly preprocessing step, we construct datasets with a spare level according to play count for songs. We classify data into ten group sparsity levels from listing records, and its level between (0.2 to 0.4), (0.4 -0.6), (0.6-0.8) and (0.8-1.0) the last one has the highest sparsity. Equation 6 calculates the sparsity level [13].

Sparsity measure = 1- (nR/ nUsers _ nItems) (6)

The symbol nR total number of play counts and nUsers number of users and nItems are a number of items or songs on the user-item matrix. We will measure the performance of our algorithm by using RMSE (root mean square error) is a predictive accurate metric. It's used widely with recommendation systems. A smaller value of RMSE suggests better performance.

RMSE is defined as follows in equation 7:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=0}(p_{i,c,t} - r_{ic,t})^2} \qquad (7)$$

Where the predicted rating for user u on an item is pu,i , the actual rating is ru,i, and the total number of ratings on the items is N.

4.2 An Experimental environment

We run our experiments on a framework machine 16 GB of RAM and Intel Core I7 CPU and windows 7. We used IntelliJIDEA software to write Java programming code to run our recommender systems. Also, we used the WEKA environment to apply association rule and clustering-based techniques like K-means in our new algorithms.

4.3 Experimental results

In this section, we run our four experimentally according to the sparsity level, we mentioned in the previous section, and we show the difference accuracy result from our two new proposed algorithms against basic collaborative filtering techniques. Table 3 shows the result of accuracy using precision, recall, and F-measure metric. Also, we need to mention that we used the merge datasets between implicit and explicit last.Fm data sets [13] to find a high accuracy level about recommended items or songs to users. According to the values in table 3, and the results are shown in figure 4,5,6,7 through the sparsity level, the accuracy with basic CF its decrease while the sparsity increased. But, our two new algorithms control the accuracy through the sparsity level against basic CF and improved by 22 % this because of the ability to find neighbors and association rule to recommend items to users and merging the implicit and explicit data (rating values to songs). The value for precision in our proposed algorithms is good because we recommend a smaller number of not matching songs to the user and using SVD as dimensionality reduction techniques and using association rule to find the hidden relations and count number of items per transactions. We have a higher precision value; it's improved by 37%. The recall based on songs not recommended to users, and it's improved in our algorithm by 10%.the F measure it's improved by 17% so that our proposed algorithm is the best in recommended songs to users.

TABLE 3

EXPERIMENTAL RESULTS

Techniques	*Sparsity from (0.2-0.4)*		
	Precision	*Recall*	*F-measure*
Basic CF	*0.54*	*0.71*	*0.61*
SVD and clustering algorithm	*0.70*	*0.66*	*0.67*
Association rule and clustering algorithm	*0.96*	*0.64*	*0.76*
Techniques	*Sparsity from (0.4-0.6)*		
	Precision	*Recall*	*F-measure*
Basic CF	*0. 63*	*0.7*	*0.66*
SVD and clustering algorithm	*0.75*	*0.67*	*0.7*
Association rule and clustering algorithm	*0.93*	*0.64*	*0.75*
Techniques	*Sparsity from (0.6-0.8)*		
	Precision	*Recall*	*F-measure*
Basic CF	*0.57*	*0.6*	*0.58*
SVD and clustering algorithm	*0.85*	*0.61*	*0.71*
Association rule and clustering algorithm	*0.95*	*0.62*	*0.75*
Techniques	*Sparsity from (0.8-1.0)*		
	Precision	*Recall*	*F-measure*
Basic CF	*0.52*	*0.53*	*0.52*
SVD and clustering algorithm	*0.88*	*0.58*	*0.69*
Association rule and clustering algorithm	*0.89*	*0.6*	*0.71*

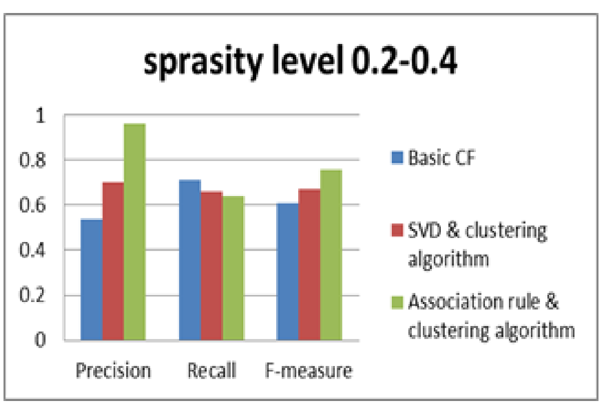


Fig. 4 experimental result while sparsity level from 0.2-0.4.

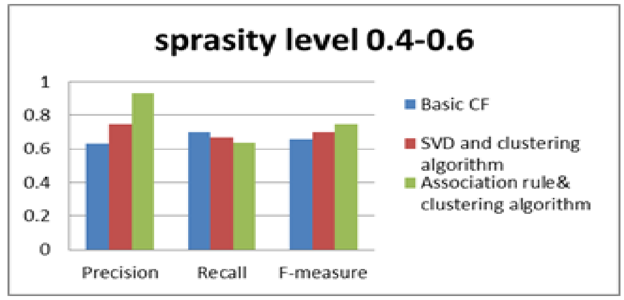


Fig. 5 experimental result while sparsity level from 0.4-0.6.

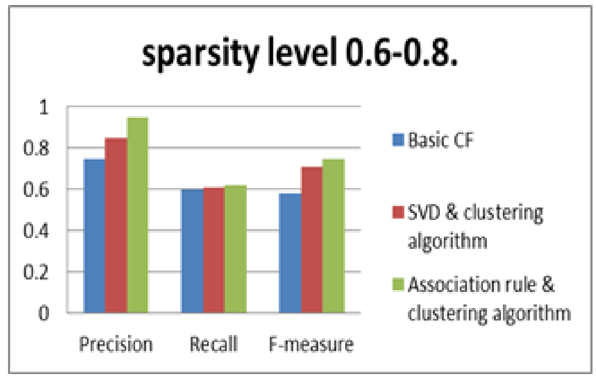


Fig. 6 experimental result while sparsity level from 0.6-0.8.

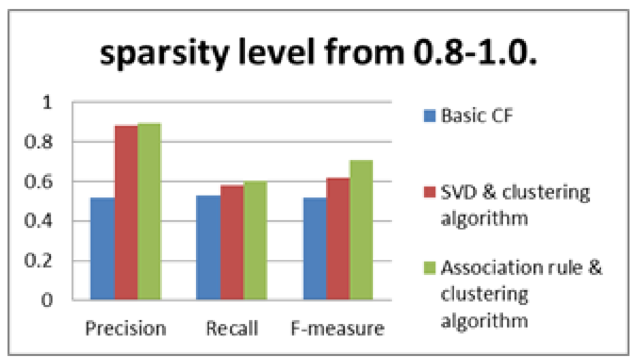


Fig. 7 experimental result while sparsity level from 0.8-1.0.

We use other datasets movie-lens data sets 1M to measure the effect of our proposed algorithms while comparing with K-means collaborative filtering techniques. Using RMSE to measure the accuracy and performance. Figure 8 shows the value of RMSE for k-means collaborative filtering compared with k-means SVD new algorithms while increasing numbers of k-nearest neighbor-based recommendation. The figure 8 shows that our new algorithm is better in accuracy rather k-means collaborative filtering techniques while grouping number of neighbors in the clusters from 10 up to 100 neighbors our new algorithm is the best in RMSE result.

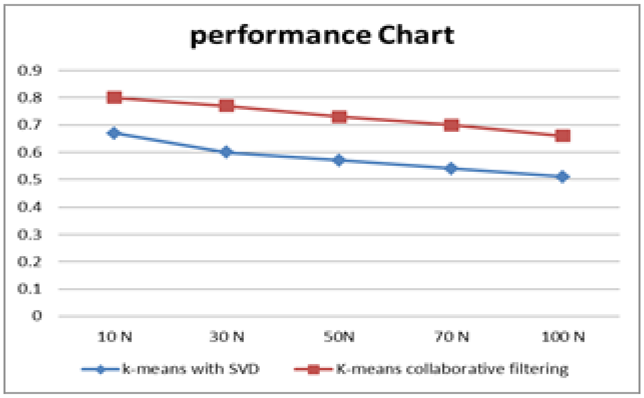


***Fig. 8 Experimental** results for performance measure.*

Our proposed algorithms advantages:

1. The ability to recommend items using user preferences not only rated items using users' item tags and a count of played songs and movies.
2. We are merging between explicit data and implicit data it's improve accuracy.
3. The ability to control accuracy while sparsity increased.
4. We use number of played counts with association rule to find hidden relationships between users.
5. We can recommend to users' songs like his preferences and ability to recommend novelty and diversity songs which, is one of a recommended items challenge.
6. Using K-means with SVD increase in performance while selecting the centroid of the cluster and using SVD to reduce the dimensionality effect in getting the result faster than traditional k-means with collaborative filtering.
7. Our two new algorithms are best then collaborative filtering techniques while used with different data sets.
8. We measure accuracy and performance in our paper and solve accuracy and sparsity recommender system challenges.

5 CONCLUSIONS AND FUTURE WORK

Recommendation systems are the most research area todays with social media and big data applications [17], information's needs to be analyzed. In this research paper, we used the million song datasets (last.Fm) it's a type of implicit data merged with explicit data users rated songs to improve the accuracy of prediction. Also, we used movie lens data sets to measure the performance and accuracy of our new algorithms. We apply our experiments on different levels of sparsity and our new algorithm improves in accuracy through the F-measure by 15% it's relevant to best accuracy result against basic collaborative filtering techniques which suffer from finding neighbors where data are spare and our algorithm also solve the novelty and diversity challenge to recommend songs to users. Using association rule mining with clustering based is better than using collaborative filtering and k-means with SVD. Our new k-means with SVD algorithms it's best in performance result compared to k-means collaborative filtering while reducing the data size using SVD in implicit data sets movie lens. In the future, what about applying these algorithms with big data sizes [18, 19] and using spark [20] to improve recommendation systems .Running new big data techniques using HDFS [21,22] to measure the performance. Also, change the type of data like books and apply our new algorithm then compare this algorithm with other recommendation systems techniques like hybrid representations, probabilistic learning[23]. Also, we need to measure the scalability of our algorithms.

6 REFERENCES

[1] Sobia Zahra, Mustansar Ali Ghazanfar, Asra Khalid, Muhammad Awais Azam, Usman Naeem, Adam Prugel Bennett, " Novel centroid selection approaches for KMeans-clustering based recommender systems ", Information Sciences Vol. 320, pp. 156-189, May 2015.

[2] Shivani Sharma," A Recommender System Based on Improvised K- Means Clustering Algorithm ", International Journal of Research in Advent Technology, Vol.6, No.7, July 2018,pp.1477-1483.

[3] Marwa Hussien Mohamed, Mohamed Helmy Khafagy, Mohamed Hasan Ibrahim . "Recommender Systems Challenges and Solutions Survey", 2019 International Conference on Innovative Trends in Computer Engineering (ITCE'2019), Aswan, Egypt, 2-4 February 2019,pp.149-155

[4] Hamdy Fadl Abdulkarem, Ghada Y. Abozaid, Mostafa I. Soliman"Context-Aware Recommender System Frameworks, Techniques, and Applications: A Survey".2019 International Conference on Innovative Trends in Computer Engineering (ITCE'2019), Aswan, Egypt, 2-4 February 2019,pp.180-185

[5] Gawesh Jawaheer, Martin Szomszor, Patty Kostkova,"Comparison of Implicit and Explicit Feedback from an Online Music Recommendation Service",(2010).doi. 10.1145/1869446.1869453.

[6] Phongsavanh Phorasim,Lasheng Yu." Movies recommendation system using collaborative filtering and k-means", International Journal of Advanced Computer Research, Vol 7(29),pp.52-59.

[7] P. Masethungh, S. Kumaresan," Centroid Selection Approaches for K-Means Clustering based Recommender Systems ",SEEE DIGIBOOK ON ENGINEERING & TECHNOLOGY, VOL. 01, MAY 2018

[8] Lu Yang, Anilkumar Kothalil Gopalakrishnan," A Collaborative Filtering Recommendation Based on User Profile and User Behavior in Online Social Networks", 2014 International Computer Science and Engineering Conference (ICSEC),pp.273-277,2014.

[9] Acilar, A. M, Arslan, A " A collaborative filtering method based on artificial immune network". Expert Systems with Applications 36, (2014),pp. 8324-8332.

[10] Hafed Zarzour, Ziad Al-Sharif, Mahmoud Al-Ayyoub, Yaser Jararweh," A New Collaborative Filtering Recommendation Algorithm Based on Dimensionality Reduction and Clustering Techniques", 9th International Conference on Information and Communication Systems (ICICS) , pp.102-106,2018.

[11] Phongsavanh Phorasim,Lasheng Yu." Movies recommendation system using collaborative filtering and k-means", International Journal of Advanced Computer Research, Vol 7(29),pp.52-59.

[12] http://millionsongdataset.com/lastfm

[13] Gunjan Advani,Neha Soni." A Novel Method for Music

Recommendation using Social Media Tags", International Journal of Computer Applications (0975 – 8887) Volume 122 – No.2, July 2015,pp.37-43.
[14] Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere."The Million Song Dataset. In Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011), 2011.
[15] F. M. Harper and J. A. Konstan, “The MovieLens Datasets,” ACM Transactions on Interactive Intelligent Systems, vol. 5, no. 4, pp. 1–19,Dec. 2015.
[16] Subhash K. Shinde ⇑, Uday Kulkarni." Hybrid personalized recommender system using centering-bunching based clustering algorithm". Expert Systems with Applications39, 2012, pp. 1381-1387.
[17] Marwa Hussien Mohamed, Mohamed Helmy Khafagy, Mohamed Hasan Ibrahim . “Hash Semi Join Map Reduce to Join Billion Records in a Reasonable Time.” Indian Journal of Science and Technology, vol. 11, no. 18, Jan. 2018, pp. 1–9., doi:10.17485/ijst/2018/v11i18/119112.
[18] Marwa Hussien Mohamed, Mohamed Helmy Khafagy, Mohamed Hasan Ibrahim . "From Two-Way to Multi-Way: A Comparative Study for Map-Reduce Join Algorithms". WSEAS Transactions on Communications, Volume 17, 2018, pp. 129-141.
[19] Marwa Hussien Mohamed and Mohamed Helmy Khafagy. “Hash semi cascade join for joining multi-way map reduce.” 2015 SAI Intelligent Systems Conference (IntelliSys)(2015): pp.355-361.
[20] Li Xie , Wenbo Zhou, Yaosen Li et al. “Application of Improved Recommendation System Based on Spark Platform in Big Data Analysis.” CYBERNETICS AND INFORMATION TECHNOLOGIES, vol. Volume 16, No 6, 2016, pp. 245–55, doi:DOI: 10.1515/cait-2016-0092.
[21] Mostafa rabea Kaseb,Mohamed helmy khafagy,ihab A.Ali,Elsayed M.Saad,"Multi-split HDFS Technique for improving data confidentiality in Big Data Replication",2019 advances in intelligent systems and computing,pp.132-142.
[22] Radya R.sahal,Mohamed helmy khafagy,fatma A.omara,"Big data multi-query optimization with apache flink",international journal of web engineering and technology,2018.
[23] Mostafa rabea Kaseb,Mohamed helmy khafagy,ihab A.Ali,Elsayed M.Saad," Redundent independent files (RIF): A technique for reducing storage and resourses in big data replication ",2018 advances in intelligent systems and computing.