IMPROVING THE ACCURACY OF BUSINESS RECOMMENDATION SYSTEM USING DISTANCE METRIC TECHNIQUES

Mustapha Maidawa[1], Aminu Ahmad[2] A. Y. Dutse[3] & Abdulsalam Ya'u Gital[4]

[1,2&3] *Department of Management and Information Technology, Abubakar Tafawa Balewa University Bauchi*

[4] Department of Mathematical Science, *Abubakar Tafawa Balewa University Bauchi*

ABSTRACT

The recommendation system is facing the problem of how to overcome sparsity, scalability and cold start issues. The existing capsule networks take time in training making it a slow algorithm. Existing literature fills up those sparse values with column or row means. as mean ignores the underlying correlation of data, it sacrifices accuracy. Therefore, to address these issues, this research proposed a hybrid collaborative base recommendation system using an improved SVD and distance metric algorithm to improve accuracy and sparsity issues. Experimental result shows that the proposed model has consistently performed better than all the three state-of-the-art methods including the Capsule Neural Network CF algorithm, the KNN CF algorithm and the SVD+SOM clustering base recommender system. This study has proven that data mining can help companies and business managers to visualize hidden patterns and trends in datasets that were not visible before. Whatever insights are revealed, they make clear decisions that benefit the company, the customers, and the stakeholders they serve.

Keyword: *Recommender System, K-Nearest Neighbor, Jaccard Distance, Euclidian Distance, and Cosine Distance.*

1. INTRODUCTION

Business Intelligence is a multidisciplinary field that encompasses technology, analytics, data management, and business strategy [1]. It plays a crucial role in enabling data-driven decision-making and helping organizations leverage their data assets to gain a competitive edge in the market. Business Intelligence (BI) refers to the technologies, applications, practices, and processes used to collect, analyze, integrate, and present business information for decision-making purposes. It involves gathering and transforming raw data into actionable insights to support strategic and operational decision-making within an organization. This is achieved using various knowledge extraction techniques [2].

Knowledge extraction, business intelligence (BI), and recommender systems are interconnected in the context of leveraging data to provide valuable insights and personalized recommendations to support decision-making and enhance user experiences [3]. Overall, knowledge extraction provides the foundation for BI by extracting insights from data, and BI, in turn, contributes to recommender systems by providing relevant insights for personalized recommendations. Together, these concepts enable organizations to extract valuable knowledge, make informed decisions, and deliver tailored recommendations to enhance user experiences and drive business growth [4]. It is in the light of these developments that this study proposed an intelligent AI framework base on recommender system for business support system.

However, recommendation system is facing the problem of how to overcome sparsity, scalability and cold start issues. The existing capsule networks take times in training making it a slow algorithm. Also, ignoring the sparsity in the datasets have result to reduction in prediction accuracy. Other existing literature fill up those sparse values with column or row means. as mean ignores the underlying correlation of data, it sacrifices accuracy. Hence, this study examined the existing framework and the need to provide a solution to the problem by proposing the inclusion of business intelligence component framework base on recommender system. Therefore, to address these issues, this

research proposed a hybrid collaborative base recommendation system using an improved SVD and self-organized map neural network (SOM) to improve cold start, accuracy, speed and sparsity issue of the existing recommendation by integrating SOM clustering for clustering the dataset, an improved SVD for dimensionality reduction and sparsity, and collaborative approach for improving the accuracy and sparsity issue.

2. RELATED

Recommender System is an application of Web Mining [5]. For example [6] proposed an efficient technique for recommender system based on Hierarchical Clustering. The results demonstrates that Chameleon based Recommender system produces less error as compared to K-means based Recommender System The research addresses the basic necessity of today's recommender system which is accuracy and speed. However, the running time of chameleon algorithms can further be reduced by using any parallel framework like map reduce. Moreso, [7] proposed a Movie Recommender System. The system has been developed in PHP and currently uses a simple console-based interface. However, testing the model on a larger data set that will enable more meaningful results.

Similarly, [8] develop a recommendation system based on users' ratings and evaluate the system using statistical evaluation techniques. The clustering accuracy was found to be good and needs less iterations to converge by the means of initial seeding. For large number of labels, the processing performance showed by Random Forest was satisfying, also when the number of clusters k were greater than 100. The proposed method has shown an improvement of .75%. thus, there is an improvement of (0.75 %) to the Softmax Regression method which is most efficient amongst others. However, the model will be impacted as the size of the dataset increases significantly.

Additionally, [9] proposed a clustering approach based on item metadata information's Evaluations are clustered according to item genre. The method improves the MAE between 0.3 and 1.8%, and the RMSE between 4.7 and 9.8%. Each cluster provides its own rating prediction and weighting strategies however, for more specific systems, the weighting strategy could be chosen according to its size and objectives.

Moreso, [10] proposed a recommender system for tourism industry using cluster ensemble and prediction and machine learning techniques. The cluster ensembles can provide better predictive accuracy for the proposed recommendation method in relation to the methods which solely rely on single clustering techniques. However, the major limitation is the scalability drawback of traditional and multi-criteria CF. hence, the need to develop the method for incremental learning and evaluate it on large multi-criteria datasets to show how it can overcome the scalability drawback.

[11] proposed nine factor analysis method, clustering technique called Genetic Weighted K-Means clustering (GWKMC) and the existing classification algorithm namely Negative Selection Algorithm (NSA). The conducted experiments confirmed the efficacy of the proposed Recommender System. Similarly, a novel web-based recommender system which is based on sequential information of user's navigation on web pages was proposed by [12]. Result clearly show that the accuracy of the proposed model is almost three times better than some existing systems. The accuracy of the proposed model is nearly 33 %. However, the research fails to include privacy, trust and social networks with the utilization of hybrid intelligent systems.

[13] proposed a deep learning neural network frame- work that utilizes reviews in addition to content-based features to generate model-based predictions for the business-user combinations. The hybrid approach is a very promising solution when compared to standalone memory-based collaborative filtering method. This methodology brings together content (user and business), collaborative (review and votes) and metadata associated with ratings under the framework of a unified supervised learning model that produces better prediction results than memory-based collaborative filtering recommendation systems. However, this research did not use geospatial data of the businesses but it can play an important role in the development of location aware recommendation systems.

Recently in 2018, A hybrid recommender system has been proposed by [5] which utilized k-means clustering algorithm with bio-inspired artificial bee colony (ABC) optimization technique. The system is novel and delivers effective fallouts when compared with already existing systems. The experiment outcomes on Movielens dataset established that the projected system provides immense achievement regarding scalability, performance and delivers accurate personalized movie recommendations by reducing cold start problem. However, the system's performance may be evaluated on advance high-configuration machine by including other important characteristics of users, such as privacy and context with cross-domain data.

Similarly, [14] proposed a clustering approach to incorporate multi-criteria ratings into traditional recommender systems effectively. Results demonstrate that the proposed approach is more accurate and effective than the traditional Pearson based collaborative filtering-based approach. Additionally, [15] proposes a novel approach called RecDNNing with a combination of embedded users and items combined with deep neural network. The experimental results on MovieLens show that the proposed RecDNNing outperforms state-of-the-art algorithms.

However, the research fails to explored more advanced deep learning methods to further enhance the quality of recommendation.

Recently in 2019, [16] proposed contextual hybrid, deep learning-based approach for session-based news recommendation that is able to leverage a variety of information types. Results confirm the benefits of considering additional types of information, including article popularity and recency. The main technical contribution of the work lies in the combination of content and context features and a sequence modelling technique based on Recurrent Neural Networks. However, outliers in the user profiles were not addressed.

Furthermore, [17] proposed MCS optimization algorithm to provide an efficient recommendation to the target user using MCFM clustering approach. The proposed MCS optimization algorithm applied on clustered data points obtained from proposed MFCM clustering performs better than other optimization algorithms. However, the research fails to implement a web-based user interface that has a user database, and has the learning model tailored to each user.

Similarly in the same year and same authors, a novel AGNN method is proposed by [18] to optimize weight of the ANN model using GA algorithm to recommend items to online targeted users using a new modified k-means approach to improve RS accuracy. The proposed RS model obtains better recommendation results over the other models used in the comparison. However, the research fails to incorporate different machine learning and clustering algorithms and study the comparative results.

Moreover, [19] proposes an Intelligent Recommender System (IRS) based on the Random Neural Network; IRS acts as an interface between the customer and the different Recommender Systems that iteratively adapts to the perceived user relevance. On average, IRS outperforms the Big Data recommender systems after learning iteratively from its customer.

More recently in 2020, [20] Uses several algorithms to obtain groupings, such as the K-Means algorithm, birch algorithm, mini-batch K-Means algorithm, mean-shift algorithm, affinity propagation algorithm, agglomerative clustering algorithm, and spectral clustering algorithm. Clustering performance evaluation shows that the K-Means method exhibits good performance with the Calinski-Harabaz Index with a score of 59.41, and the birch algorithm with a score of 1.24, on the Davies–Bouldin Index. However, the research does not implement a web-based user interface that has a user database.

3. METHOOLOGY

This research proposes introduces a novel method for creating a hybrid recommender framework that combines Collaborative Filtering (SVD) with Self-Organizing Map neural network technique for knowledge extraction in relational and non-relational datasets. The new chart (research Model) has the following key component: knowledge base, learning module, clustering, classification, and decision manager.

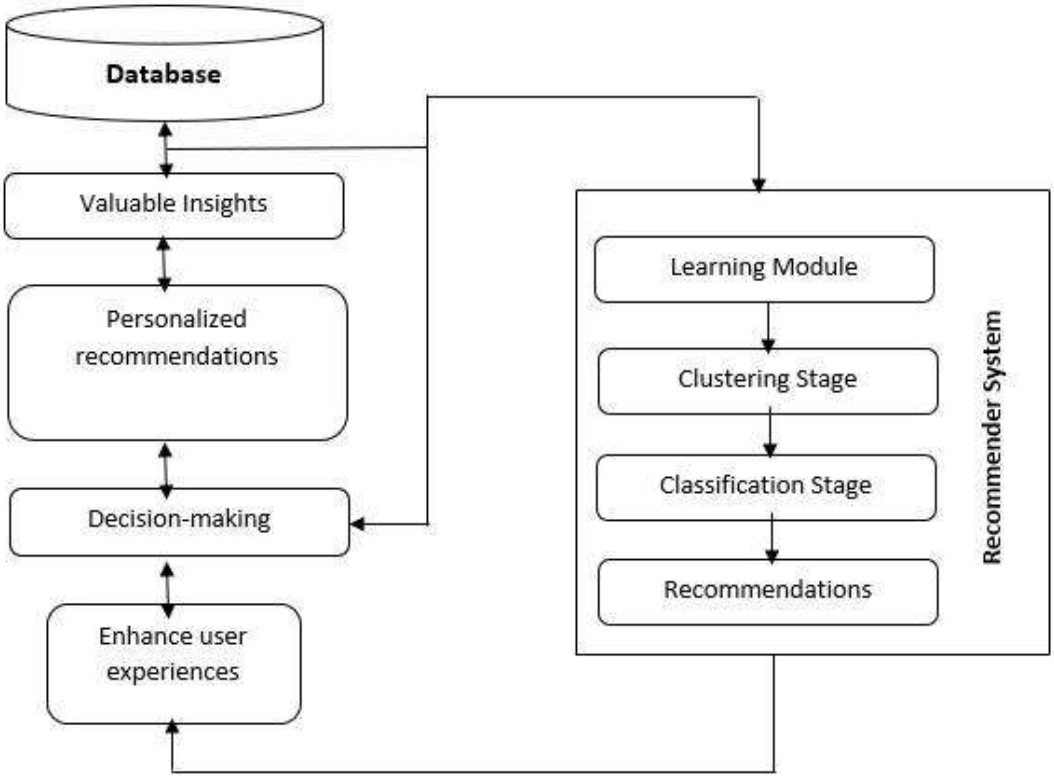

Fig. 1. System architecture

There is a lot of information in this business world. We need to maintain information for decision making in the business environment. Decision making consists of two types of data: online analytical processing (OLAP) and online transaction processing (OLTP). The former contains historical data about the business from the beginning, and the latter contains only daily transactions about the business. Based on these two types of data, we use a novel method for creating a hybrid recommender framework that combines Collaborative Filtering with Self-Organizing Map neural network technique and user knowledge to improve business intelligence based on frequent item set mining and clustering. We can run the decision-making process. This model consists of three main phases: data preprocessing base on three main distance metric approach to create an improved SVD for filling data sparsity, followed by clustering, classification, and finally the recommendation phase is constituted.

A. Data Sparsity Preprocessing with Cosine, Jaccard and Euclidean Distance

On and above this, one serious drawback of the existing recommendation approaches is the fact that they do not give insights into the rating patterns. For example, after looking into several ratings from a user one cannot get an understanding on what might have motivated the user to give such ratings. We believe that the singular value decomposition (SVD) of the user rating matrix will give us insights into such underlying concepts or patterns. However, the normal SVD does suffer from a drawback that it does not perform well when there are missing values in the rating matrix.

Also, Recommender systems in real world deals with a large amount of sparsity or unobserved value. Since zero values have certain meaning in case of rating matrix, estimating the missing values to zero provides wrong predictions. Thus, an alternative approach would be to use some prediction algorithm for imputation of missing values. The most common approach is to fill up those values with column or row means. The full matrix thus obtained can be then used for SVD calculation although an efficient way would be to re-center the matrix before proceeding with the SVD calculations. However, as mean ignores the underlying correlation of data, it sacrifices accuracy.

To overcome the aforementioned issues, we have pre-processed the data. At first, we have used three main distance metrics to identify the K-nearest neighbors of a given user and then based on their ratings we have filled up the missing rating values of the user in question.

The main aim of this phase is to improve the performance of SVD algorithm. For this purpose, we calculated the three different distance metrics namely, cosine, Jaccard and Euclidean.

I. Cosine Distance Metric

This distance metric is used mainly to calculate similarity between two vectors. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in the same direction. It is often used to measure document similarity in text analysis. When used with KNN this distance gives us a new perspective to a business problem and lets us find some hidden information in the data which we didn't see using the above two distance matrices.

It is also used in text analytics to find similarities between two documents by the number of times a particular set of words appear in it. Formula for cosine distance is:

$$\cos\theta = \frac{\vec{a} \cdot \vec{b}}{\|\vec{a}\| \cdot \|\vec{b}\|} \quad \ldots(1)$$

Using this formula, we will get a value which tells us about the similarity between the two vectors and 1-cosΘ will give us their cosine distance. Using this distance, we get values between 0 and 1, where 0 means the vectors are 100% similar to each other and 1 means they are not similar at all.

II. Jaccard Distance Metric

The Jaccard similarity index (also known as the Jaccard similarity coefficient) is used to compare members of two sets to find distinct and shared members in them. Its value ranges from 0% to 100% and signifies the similarity between the two sets under consideration. The Jaccard coefficient is a similar method of comparison to the Cosine Similarity due to how both methods compare one type of attribute distributed among all data. The Jaccard approach looks at the two data sets and finds the incident where both values are equal to 1. So, the resulting value reflects how many 1 to 1 match occur in comparison to the total number of data points. This is also known as the frequency that 1 to 1 match, which is what the Cosine Similarity looks for, how frequent a certain attribute occurs.

It is extremely sensitive to small samples sizes and may give erroneous results, especially with very small data sets with missing observations. The formula for Jaccard index is:

$$\mathbf{D(A,B) = 1 - J(A,B)} \quad \ldots(2)$$

III. Euclidean Distance Metric

The Euclidean distance or Euclidean metric is measured as the distance between two points in the Euclidean space. This distance is the most widely used one as it is the default metric that SKlearn library of Python uses for K-Nearest Neighbour. It is a measure of the true straight-line distance between two points in Euclidean space. It can be used by setting the value of p equal to 2 in Minkowski distance metric.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \quad \ldots(3)$$

Thus, in this research, the three distance matrices were used to find K-nearest neighbors from the fast KNN library. The objectives are to propose a method for improving the performance of SVD matrix in the context of recommender systems. We will test the approach for a range of values of K for which the performance of the improvised method is significantly better than the normal SVD.

B. Clustering Technique

After the clustering stage, similarity calculation is performed on the selected cluster. This similarity measure calculates the similarity between users/items. The result will be used to predict the ratings of a missing values using the KNN base approach. This is with a view to reduce the sparsity of the ratings and improve the accuracy of the result. thereby the improved SVD and self-organizing map neural network clustering technique is applied on the data sets to recognize and group the instances in the datasets. The SOM, SVD and KNN technique distributes input vectors into separated clusters by means of similarity and distance measurement. All input vectors are assembled into distinct centers by means of minimizing objective function. Dimensionality reduction such as SVD will be performed on the output of the similarity measures to produce ranked list of items. It is performed on past ratings in order to get the target user/item similarities to other users/items.

In the classification phase, collaborative filtering algorithm base on neural network (SOM) architecture is applied to instances of the database to extract the knowledge learned. The training data sets are separately classified using each classifier to create a model. The testing data sets are then predicted based on the created models. Finally, the last phase involves generating prediction as well as recommendation for the target user. the system generates recommendation base on the frequency of predicted items.

i. Collaborative filtering model

Collaborative filtering (CF) is a method for providing suggestions based on correlations between users and products. In other words, it is the method of filtering items based on the opinions of other users and choosing a group of users with similar tastes to a specific user. The method analyzes their favorite products and integrates them into a categorized list of suggestions. The CF system tries to find similar items based on user feedback. User feedback can be either explicit or implicit, explicit as a numerical rating to specify how much users liked a particular item, for example 1 means dislike or 5 if the user likes the item very much, or implicit like browsing history on the website or reading time a type of product. Collaborative Filtering have two types of algorithms, Memory-Based and Model-Based, the first type saves products and user data in memory, then uses mathematical methods to make estimates based on the data. Different machine learning algorithms, such as the Bayesian network, rule-based, and clustering methods, are used to construct the model process. In our approach we will use the second category, Model-Based, because it can response user's request instantly. Singular Value Decomposition (SVD) is the approach we used to create a Collaborative Filtering model. SVD is a matrix factorization technique that reduces the number of features in a dataset by decreasing the space dimensions from A to B, where A is smaller than B. Since we are interested in the matrix factorization aspect of recommendation structures that maintains the same dimensionality, the key function of SVD is to decompose a matrix into three other matrices.

ii. Neural network model

Neural networks are a group of algorithms that detect patterns and are closely modeled after the human brain. They use a kind of machine vision to classify sensory data, naming or sorting raw data. Both real-world relational and non-relational database records, whether images, sound, text, or time series, must be converted into the patterns they understand, which are digital and stored in vectors. We should think of neural networks as a layer of clustering and classification on top of the data we store and handle, as they assist us in clustering and classifying data. The self-organized map is a kind of neural network, and it is the technique we will be using in this research to address the limitations of capsule neural network.

As explained in Collaborative Filtering module, CF is the process of filtering items based on users' historical opinions and preferences on a set of items. Here, we will use the Self-Organizing Map (SOM) method to improve the traditional collaborative filtering system in order to build our hybrid system. A self-organizing map (SOM) is a form of artificial neural network (ANN) that is trained using unsupervised learning to generate a low-dimensional, usually two-dimensional. We will use the self-organizing map in our model to solve the issue of unsupervised clustering of the movie dataset. Clustering technology simplifies the structure of the dataset and divides it into different clusters, so the users can easily observe and analyze the data. After getting our user's favorite class, we use the SVD collaborative filtering approach to predict items ratings and sort them from best to worst to give recommendations. The algorithm for the proposed method is presented below.

Algorithm 1: The Proposed hybrid SVD-KNN base SOM Recommender System

Input: User-Item rating matrix A, User-Item Context

Output: Recommendation

Step 1: Input user item matrix including users' ratings data;
Step 2: Check Sparsity and input missing data based on new Jaccard & Euclidian base SVD Algo.
Step 3: Create user clusters using clustering SOM
Step 4: Calculate similarity of each cluster & compute the rating prediction.
Step 5: Use SVD to obtain the decomposition matrices of each cluster;
Step 6: For each matrix obtained from the decomposition step, apply context (user or item context) and calculate the similarity;
Step 7: Output Recommendations
Step 8: Evaluate Recommender system using MAE, RMSE, ACC. PREC. & REC.

C. Evaluation and Test

The proposed hybrid algorithm provides a solution for the recommender system. this methodology can be validated through various experimental settings. This assignment uses a several datasets from ecommerce industry set to test a proposed algorithm to encourage consumers to buy a product. This algorithm helps active users find the

items they want to buy from their business. E-commerce businesses that use the recommendation system are Amazon.com, CDNOW.com, Drugstore.com, eBay, MovieFinder.com and Reel.com.

The offline evaluation method for in -domain recommender systems is an easier way to evaluate recommender systems. In this method, the data set containing user information, items and evaluations is divided into a training set and a validation set, a model training set and a model test validation set. System performance is further evaluated with a set of validations using different evaluation techniques. The offline evaluation method is the easiest to use because it provides an opportunity to weigh the recommended algorithms differently from each other. Thus, the performance of the proposed algorithm can be compared with the existing methodology using various metrics such as accuracy, recall and precision. This can be done by looking for qualities in different numbers of neighbors, iterations and clusters to ensure that the performance of the proposed method is better than the existing method.

The accuracy of the rating prediction was measured using Five (5) commonly used evaluation indicator namely Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Recall, F-metric, and Precision. The MAE and RMSE were obtained by calculating the difference between the true rating and the predicted rating. A smaller value corresponds to higher accuracy of the rating prediction. MAE can be calculated by the following formula.

$$MAE\,(Pred\,,act) = \sum_{i=1}^{N} \left| \frac{Pred_{u,i} - act_{u,i}}{N} \right| \quad \ldots(4)$$

The precision measures the portion of item that are relevant within the received result. While the F-score or F-measure is a measure of a test's accuracy. Both metrics should be used in common. Precision and F-score can be calculated with the following formulas.

Precision: precisions provide information about how precise/accurate your model is out of those predicted positives, how many of them are actual to positives. Precisions is a good measure to determine when cost of false positives is high. It is mathematically expressed as:

$$\text{Precision} = \frac{TP}{TP+FP} \quad \ldots(5)$$

Recall attempts to answer what proportion of actual positives was identified correctly. It is expressed mathematically as

$$\text{Recall} = \frac{TP}{TP+FN} \quad \ldots(6)$$

F-Measure is a function of precision and recall, it might be better to use when we seek to balance between precision and recall and there is an even class distribution (large number of actual negatives). It is mathematically express as:

$$F1 = 2* \frac{Precision * Recall}{Precision + Recall} \quad (7)$$

4. RESULT

The proposed recommender system was developed using Anaconda and Jupyter Notebook and run-on Intel® Pentium Duo Core 1.6GHz system having a memory of 8GB, 500GB HDD and Windows 10 64 bit. The proposed recommender system is evaluated using MovieLens 100k and 1M real world dataset and other e-commercial database such as "Amazon Instant Video (AIV)", and the other is "Apps for Android (AA). The proposed recommender system is compared against state-of-the-art 3 classical model-based CF algorithms. They include, the Capsule Neural Network CF algorithm, the KNN CF algorithm and the EM clustering base recommender system. As stated earlier, the accuracy, F-score, precision and recall are measured and evaluated in relation to other state-of-the-art methods.

To evaluate the proposed method, we evaluate the performance of the model against the MovieLens world datasets, secondly, the required libraries were imported into the python implementation environment and the dataset fed into the system. Table 1 depict the features of the movielens database comprising of movie, rating and user features as computed in the python environment.

Table 1 Features of the MovieLens database

	userId	movieId	rating	timestamp	title	genres
0	1	31	2.5	1260759144	Dangerous Minds (1995)	Drama
1	7	31	3.0	851868750	Dangerous Minds (1995)	Drama
2	31	31	4.0	1273541953	Dangerous Minds (1995)	Drama
3	32	31	4.0	834828440	Dangerous Minds (1995)	Drama
4	36	31	3.0	847057202	Dangerous Minds (1995)	Drama
...	...	...	...	...	...	...
99999	664	64997	2.5	1343761859	War of the Worlds (2005)	Action|Sci-Fi
100000	664	72380	3.5	1344435977	Box, The (2009)	Drama|Horror|Mystery|Sci-Fi|Thriller
100001	665	129	3.0	995232528	Pie in the Sky (1996)	Comedy|Romance
100002	665	4736	1.0	1010197684	Summer Catch (2001)	Comedy|Drama|Romance
100003	668	6425	1.0	993613478	6th Man, The (Sixth Man, The) (1997)	Comedy

100004 rows × 6 columns

From Table 1 The dataset contains: 100004 ratings of 9125 movies. The distribution of the datasets is shown in Fig. 2

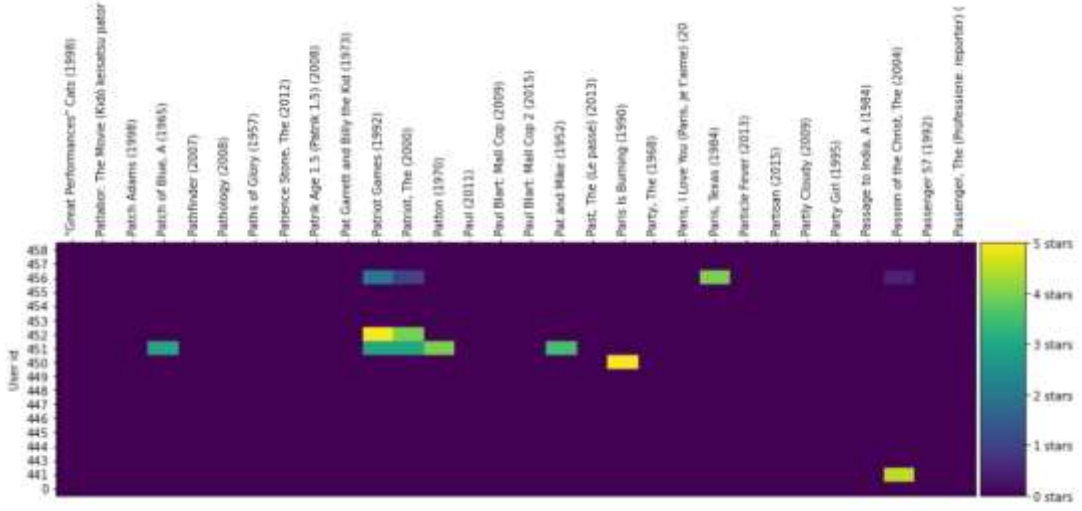

Fig. 2 Distribution of features and ratings in the movie datasets

From fig. 2 Each column is a different movie. Each row is a different user. The cell's color is the rating that each user has given to each film. The values for each color can be checked in the scale of the right.

We Create an instance of KNN to find three clusters as shown in Fig. 4.3.

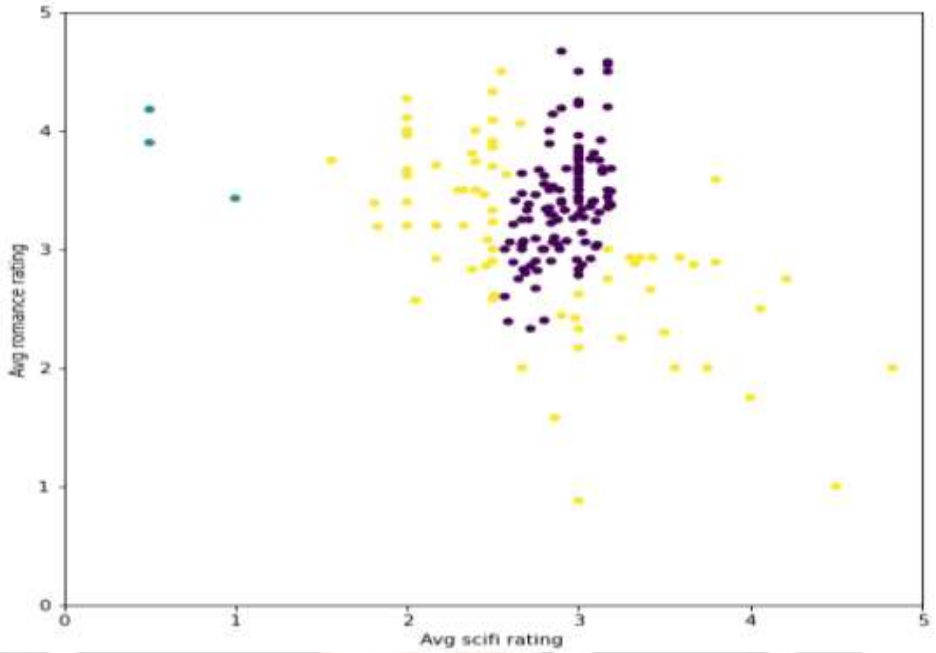

Fig. 3 instance of three clusters for avg romance rating vs avg scifi rating

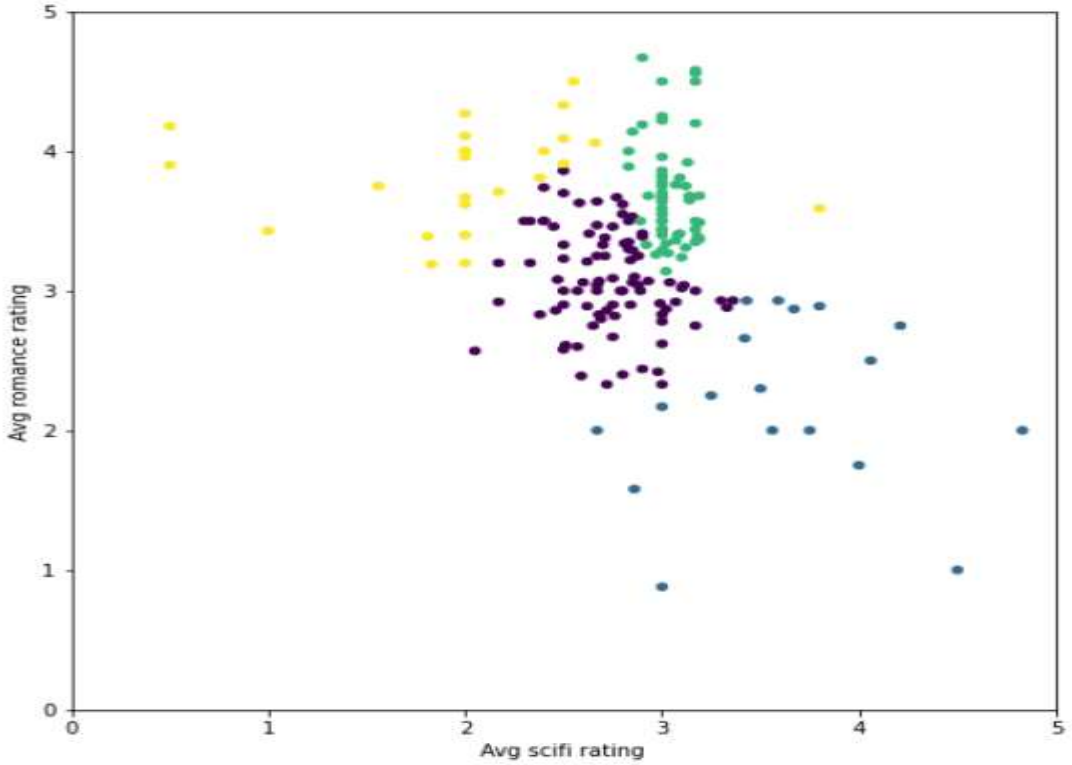

Fig. 4 Instance of four clusters for avg romance rating vs avg scifi rating

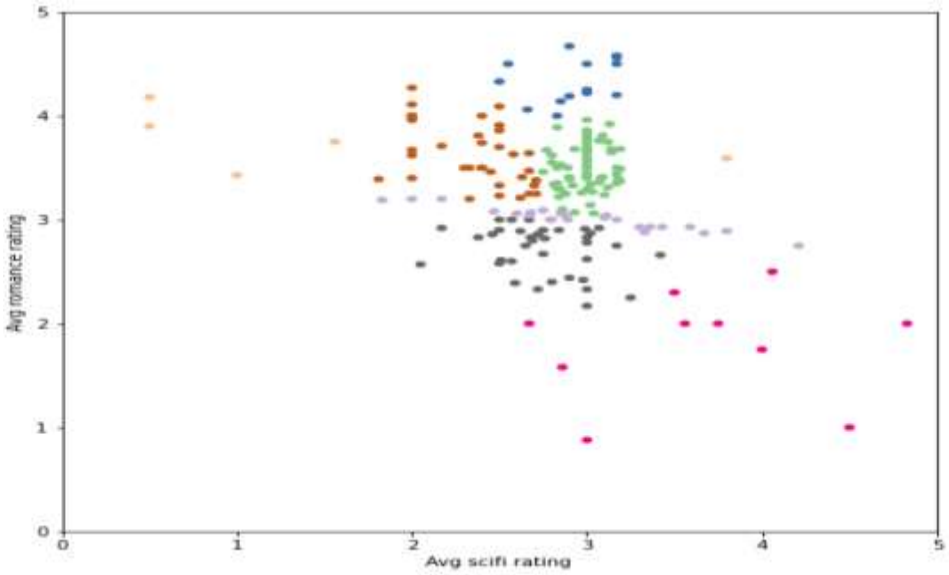

Fig. 5 Instance of seven clusters for avg romance rating vs avg scifi rating

However, the database contains missing features as shown in Fig. 6 which are relevant to the knowledge mining in business intelligence.

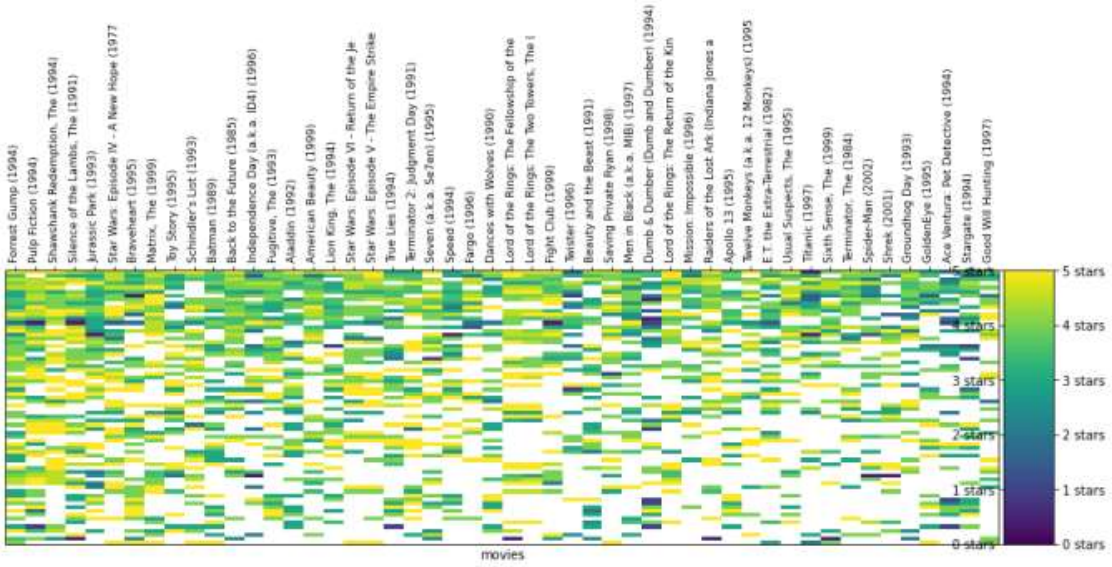

Fig. 6 Distribution of data for Movie Lens Database

From fig. 6 Each column is a different movie. Each row is a different user. The cell's color is the rating that each user has given to each film. The values for each color can be checked in the scale of the right. The white values correspond to users that haven't rated the movie. Sparsity of a dataset is defined as the ratio of unspecified ratings to the total number of entries in the user-item matrix. Therefore, we need to address the sparsity in the database first. To achieved this objective, first we calculate the sparsity level in database as shown in Table 3

Table 3 Sparse instances from the MovieLens database

title	"Great Performances" Cats (1998)	$9.99 (2008)	'Hellboy': The Seeds of Creation (2004)	'Neath the Arizona Skies (1934)	'Round Midnight (1986)	'Salem's Lot (2004)	'Til There Was You (1997)	'burbs, The (1989)	'night Mother (1986)
userId									
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
5	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
6	NaN	NaN	NaN	NaN	NaN	NaN	NaN	4.0	NaN

	0	1	2	3	4	5	6	7	8	9	...	990	991	992	993	994	
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	I
1	3.0	4.0	NaN	3.0	NaN	4.0	NaN	NaN	4.0	5.0	...	NaN	NaN	NaN	NaN	NaN	I
2	5.0	4.5	5.0	3.0	NaN	NaN	NaN	NaN	3.0	NaN	...	NaN	NaN	NaN	NaN	NaN	I
3	5.0	5.0	NaN	NaN	5.0	5.0	NaN	NaN	NaN	5.0	...	NaN	NaN	NaN	NaN	NaN	I
4	4.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	I
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
666	4.0	5.0	NaN	NaN	NaN	4.0	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	I
667	NaN	5.0	4.0	5.0	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	I
668	NaN	NaN	NaN	NaN	5.0	3.0	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	I
669	NaN	NaN	5.0	5.0	NaN	NaN	4.0	4.0	5.0	NaN	...	NaN	NaN	NaN	NaN	NaN	I
670	5.0	4.0	5.0	NaN	5.0	NaN	4.5	5.0	NaN	5.0	...	NaN	NaN	NaN	NaN	NaN	I

671 rows × 1000 columns

As stated earlier, recommendation system is facing the problem of how to overcome sparsity and cold start issues. Many researches were conducted in order to overcome the above-mentioned problem. But there is the need for improvement. Having calculated the sparsity of the four different business recommendation databases in Table 3.

Table 3 Sparsity Level in the Distributed Databases

Database	Sparsity Level
MovieLens	98.3%
1M real world	91.23
Apps for Android (AA)	92.11
Amazon Instant Video (AIV)	89.44

Base on the analyses in Table 3. It was found that the sparsity level is high 98.35% for the case of the MovieLens datasets. It is evident that there are a lot of 'NaN' values as most of the users have not rated most of the movies. This type of datasets with a number that is high of 'null' values are called 'sparse datasets. This research aim is with a view to reduce sparsity of the dataset by predicting the missing values using the an improved SVD+KNN data preprocessing.

To address this problem, in this research, we fill the missing values based on what K-nearest neighbors of the user have done. The neighbors were identified using the KNN approach baase on uclidean and jaccard distance metrics. We believe that our approach can give better results compared to the normal SVD with imputation of user's average. Thus, we employ a new method that combines the KNN base approach on SVD for data preprocessing. thus, we pre-processed the data with Jaccard distance and Euclidian distance in terms of rating matrix. The aim here is to improve the performance of SVD algorithm using two different distance metrics namely, Jaccard and Euclidean. In each case, the information of the nearest neighbors was utilized to remove the sparsity of the rating matrix to some extent. For this purpose, mean and mode of the ratings given by the nearest neighbors were used to fill up the empty spaces in the original rating matrix resulting as shown in Table 4.

Table 4 Predicted ratings of a missing values

	Pulp Fiction (1994)	Jurassic Park (1993)	Star Wars: Episode IV - A New Hope (1977)	Shawshank Redemption, The (1994)	Toy Story (1995)	Silence of the Lambs, The (1991)	Matrix, The (1999)	Schindler's List (1993)	Star Wars: Episode V - The Empire Strikes Back (1980)	Forrest Gump (1994)
3	5.0	3.0	5.0	2.0	2.0	5.0	5.0	4.0	5.0	1.0
8	5.0	4.0	4.0	5.0	4.0	4.0	3.0	5.0	4.0	5.0
18	3.5	3.0	4.0	4.0	4.0	3.0	4.0		4.5	3.5
4	5.0	4.0	4.0	4.0	3.0	3.0		4.0	5.0	5.0
33	5.0	5.0	5.0	5.0	4.0		4.0		4.0	5.0

5 rows × 300 columns

Our system classifies movies using SOM based on genre of the movies as shown in Table 4.5 After initialization of our map dimensions and randomly initialization SOM weights we train the model. Once the map has been trained, it's given us weights as results, then we use those weights as input data of KNN base SVD clustering model. we generate Recommendation for the users as shown Table 5.

Table 5. Recommendations made by the proposed system for Movie database

	movieId	title
0	34	Babe (1995)
1	309	Red Firecracker, Green Firecracker (Pao Da Shu...
2	1	Toy Story (1995)
3	337	What's Eating Gilbert Grape (1993)
4	2	Jumanji (1995)
5	145	Bad Boys (1995)
6	378	Speechless (1994)
7	3	Grumpier Old Men (1995)
8	517	Rising Sun (1993)
9	129	Pie in the Sky (1996)
10	4	Waiting to Exhale (1995)
11	312	Stuart Saves His Family (1995)
12	291	Poison Ivy II (1996)
13	5	Father of the Bride Part II (1995)
14	344	Ace Ventura: Pet Detective (1994)
15	308	Three Colors: White (Trzy kolory: Bialy) (1994)
16	6	Heat (1995)
17	20	Money Train (1995)
18	343	Baby-Sitters Club, The (1995)
19	7	Sabrina (1995)
20	519	RoboCop 3 (1993)
21	215	Before Sunrise (1995)
22	8	Tom and Huck (1995)
23	9	Sudden Death (1995)
24	270	Love Affair (1994)
25	154	Beauty of the Day (Belle de jour) (1967)

Now let's evaluate the performance of the model. we will check how well the hybrid modeling performed in terms of KNN+SVD+SOM and SVD+SOM. we evaluate the performance of all the methods at k=30. Table 6 depict the emphatical results achieved in terms of MAE, RMSE, Precision, recall and F-score for the correct prediction using user-based similarity at K=30.

Table 6: Performance achieved by all methods for K=30

Method	MAE	RMSE	Precision	Recall	F-Score
Euclidian improvised SVD+SOM	0.7740	0.9916	0.7108	0.6795	0.6947
Jaccard improvised SVD+SOM	0.7298	0.9425	0.7453	0.7198	0.7323
Cosine Improvised SVD+SOM	0.7126	0.9241	0.7568	0.7242	0.7551

From Table 6. it is noticed that at large value of k=30 cosine's improvised model has demonstrated significant improvement with superior performance over the Jaccard's and Euclidean distance. From the result the proposed cosine base SVD model attained the best and most stable predictive accuracy (MAE and RMSE) and decision

support accuracy (Precision, Recall and F-Score). It was also noticed that there is general improvement across all the three methods proposed in the study at larger k values of 30. This result can better be analyzed using the following figures presented below.

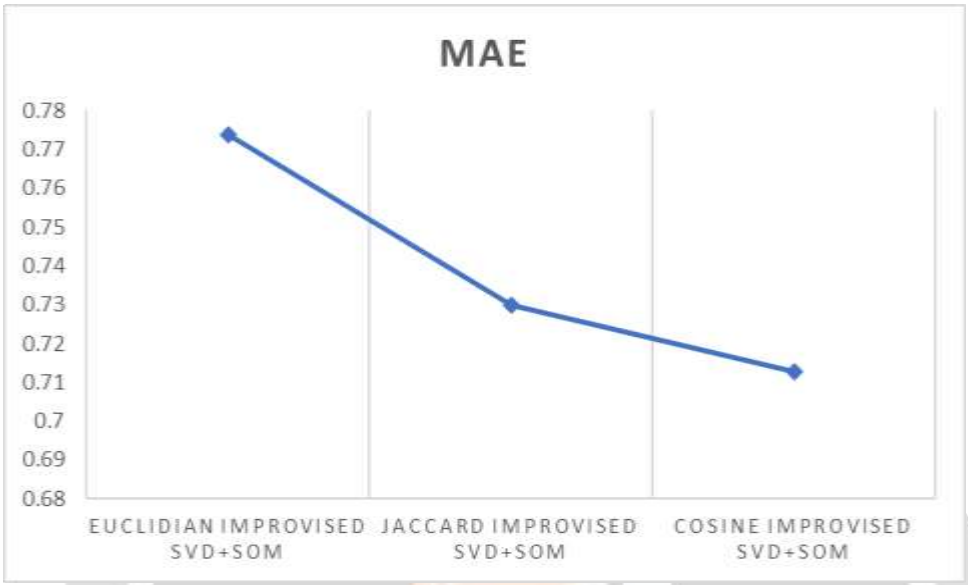

Figure 7: MAE for all three methods at k=30

As sated earlier, the MAE was obtained by calculating the difference between the true rating and the predicted rating. A smaller value corresponds to higher accuracy of the rating prediction. Thus, for the MAE, lower values indicate less errors and better prediction accuracy. From Figure 7. it is seen that the cosine's distance model attained the best and most stable MAE by achieving the lowest MAE of 0.7126 followed by the Jaccard's distance model which has 0.7298. The Euclidean distance model was the least performing model by attaining highest errors of 0.7740.

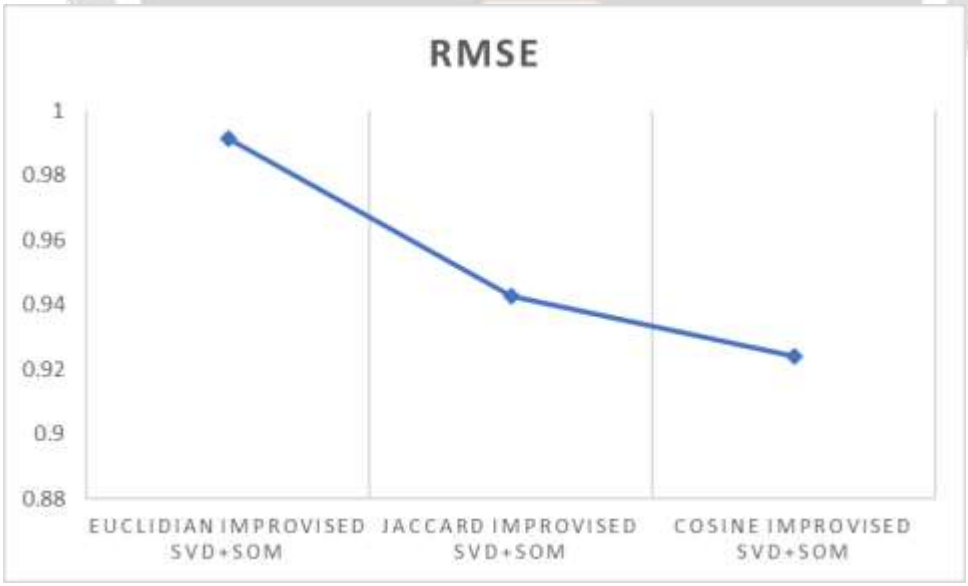

Figure 8: RMSE for all three methods at k=30

Similarly, the RMSE was obtained by calculating the difference between the true rating and the predicted rating. A smaller value corresponds to higher accuracy of the rating prediction. Thus, for the RMSE, lower values indicate less errors and better performance by the model. From Figure 8. it is seen that the cosine's distance model performs

better by attaining the least error RMSE of 0.9241 followed by the Jaccard's distance model which has 0.9425. The Euclidean distance model was the least performing model by attaining highest errors of 0.9916.

Thus, from the analysis, it is easier to say that, at K=30, the Cosine-SVD model performed better than all the three methods used in the study in terms of the predictive accuracy (MAE and RMSE). However, to further ensure the generalization of the prediction performance by each model, we further evaluate the proposed models using decision support accuracy such as precision, recall and F-score as presented in the following figures. For the decision support accuracy measures (Precision, Recall and F-score), it plays an important role for the multi-criteria recommender evaluations. Many metrics for this purpose are well known from the information retrieval area. The precision here measures the portion of items that are relevant within the received result. F-measure is a measure of a test's accuracy. For this research, the precision recall and F1 scores were reported within the range of 0-1. Higher values close to 1 means better decision support accuracy and lower values close to 0 means poor decision support accuracy. For better understanding and more intuitive discussion for decision support accuracy, the results are plotted in Fig. 9

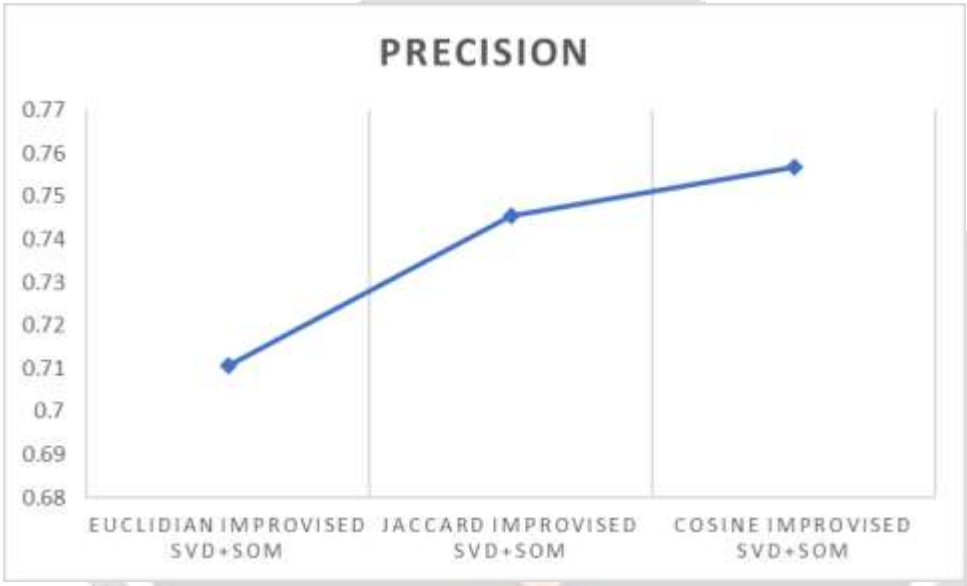

Figure 9: Precision for all three methods at k=30

As stated earlier, the precision measures the portion of item that are relevant within the received result. Precisions provide information about how precise/accurate your model is out of those predicted positives, how many of them are actual to positives. Precisions is a good measure to determine when cost of false positives is high. For this research, the precision recall and F1 scores were reported within the range of 0-1. Higher values close to 1 means better precision accuracy and lower values close to 0 means poor precision accuracy. From Figure 9. the Cosine base SVD model attain the highest score of 0.7568. however, the Jaccard base SVD came second by attaining precision score of 0.7453 while the least performing was the conventional Euclidean distance base SVD model which attains precision score of 0.7108. Similarly, the performance in terms of recall is depicted in Figure 10.

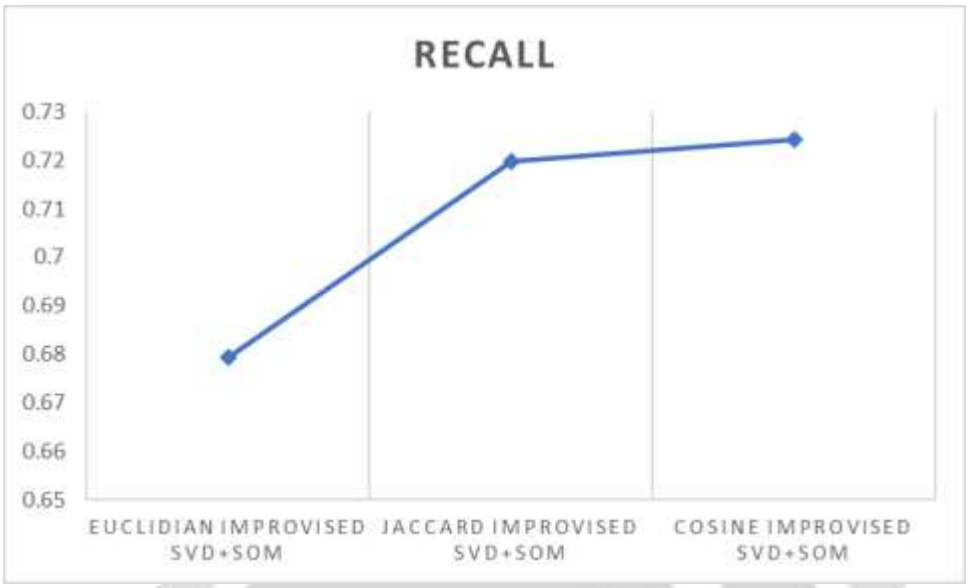

Figure 10: Recall for all three methods at k=30

As stated earlier, Recall attempts to answer what proportion of actual positives was identified correctly. For this research, the recall scores were reported within the range of 0-1. Higher values close to 1 means better recall accuracy and lower values close to 0 means poor recall accuracy. From Figure 11. The Cosine base SVD model attain the highest score of 0.7242. however, the Jaccard base SVD came second by attaining precision score of 0.7198 while the least performing was the Euclidean distance base SVD model which attains precision score of 0.6795. Finally, the performance in terms of F-score is depicted in Figure 11.

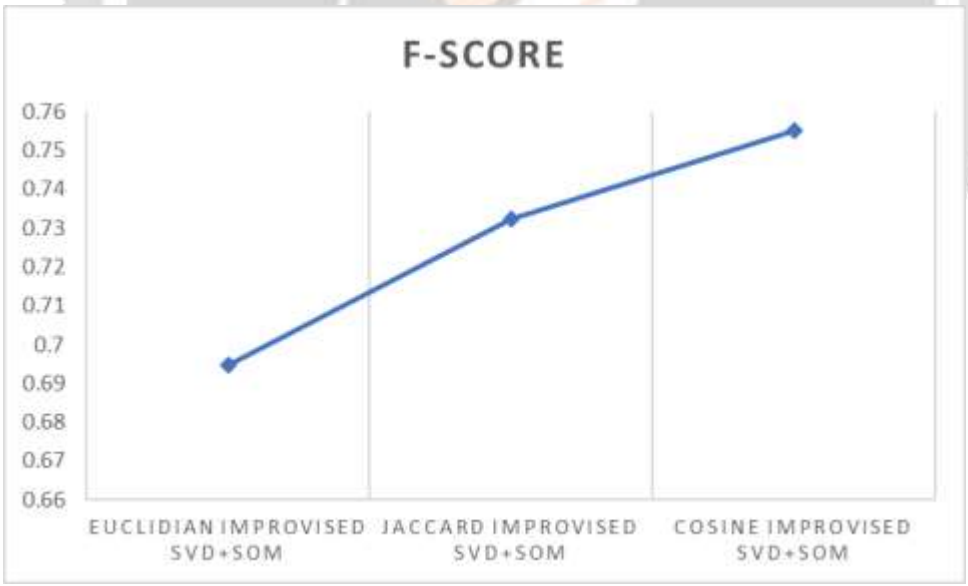

Figure 11: F-score for all three methods at k=30

As sated earlier, the F-Measure is a function of precision and recall, it might be better to use when we seek to balance between precision and recall and there is an even class distribution (large number of actual negatives). Thus, in this research, F-measure is a measure of a test's accuracy and was reported within the range of 0-1. Higher values close to 1 means better F-score and lower values close to 0 means poor F-score. From Figure 11. the Cosine base SVD model attain the highest F-score score of 0.7551. however, the Jaccard base SVD came second by attaining F-score of 0.7323 while the least performing was the conventional Euclidean distance base SVD model which attains F-score of 0.6947. The summary for the performance of the proposed model across all the five metrics used in this study at k=30 is further depicted in figure 12 for easy comparison.

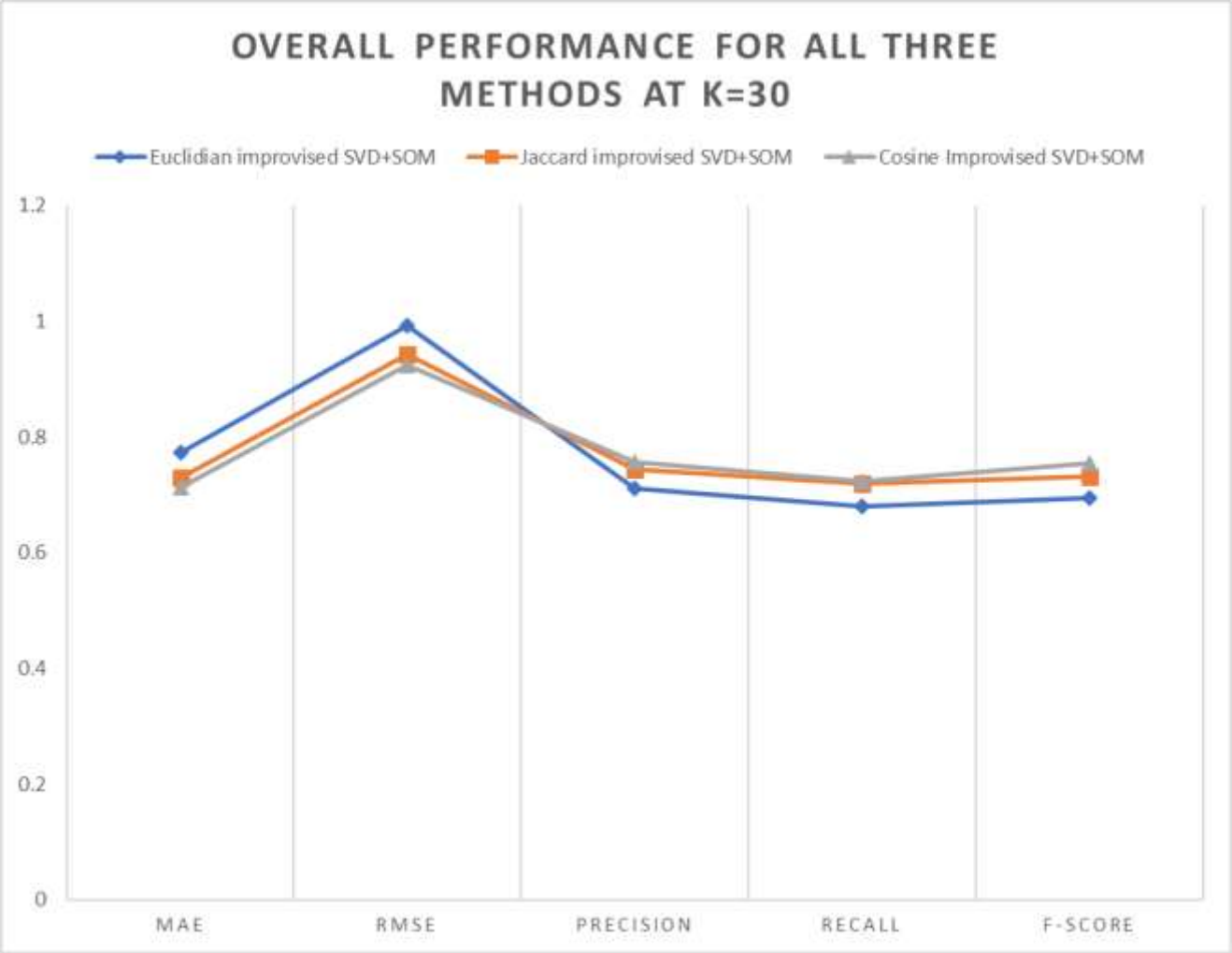

Figure 12: Overall Performance for all three methods at k=30

From Fig. 12, the proposed cosine base SVD model attains the better performance in terms of all the five metrics (MAE, RMSE precision, recall and F-score) at larger values of K= 30. This was followed by the Jaccard Improvised SVD+SOM which attains second best in terms of (MAE, RMSE precision, recall and F-score). The Euclidean base SVD model was the least performing model at large values of K. Thus, as seen in the literature, the Jaccard and Euclidean distance metrics are extremely sensitive to small samples sizes and may give erroneous results, especially with very small data sets with missing observations. Therefore, the cosine distance metrics performs better even with larger data samples and larger k values.

Therefore, the significant improve witness by the cosine distance metric improvisation to the SVD model can be attributed to the fact the Cosine Distance distance metric is used mainly to calculate similarity between two vectors. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in the same direction. It is often used to measure document similarity in text analysis. When used with KNN this distance gives us a new perspective to a business problem and lets us find some hidden information in the data which we didn't see using the above two distance matrices (Jaccard and Euclidean). It is also used in text analytics to find similarities between two documents by the number of times a particular set of words appear in it.

4. CONCLUSIONS

This research proposed a hybrid collaborative base recommendation system using an improved SVD and self-organized map neural network (SOM) to improve accuracy and sparsity issue of the existing recommendation by integrating SOM clustering for clustering the dataset, an improved SVD for dimensionality reduction and sparsity, and collaborative approach for improving the accuracy and sparsity issue. The distance matrices were used to find K-nearest neighbors using the KNN algorithm from the fast KNN library. The objectives are to propose a method for improving the performance of SVD matrix in the context of recommender systems. We apply the proposed model in different recommendation datasets and scenarios such as social recommendations, joint-recommendation, group recommendation, etc.

We test the approach for a range of values of K for which the performance of the improvised method is significantly better than the normal SVD. Experimental result shows that, the performance of the proposed improvised SVD data imputation using two different distance metrics namely, cosine, Jaccard and Euclidean. the experimental result shows that, by using both cosine and Jaccard distance to improvised the SVD, the error has been further reduced compare to the conventional Euclidean SVD approach. Jaccard and Cosine distance base data preprocessing on the SVD has significantly improve on the general performance of the normal SVD algorithm in terms both predictive accuracy and decision support accuracies respectively. This result can be attributed to the fact that the improvised SVD preprocessing approach has significantly addressed the existing data sparsity issue of recommender system to some extend as against the conventional SVD preprocessing approach. This makes it easy for the SOM to cluster and generates the recommendations to the users with high precision and accuracy.

5. ACKNOWLEDGEMENT

I wish to appreciate the expert support of my supervisors Prof. A. Y. Dutse, Prof. Aminu Ahmad & Dr. Abdulsalam Ya'u Gital for their immense contribution towards the success of this research.

6. REFERENCES

1. Liang, T.-P. and Y.-H. Liu, *Research landscape of business intelligence and big data analytics: A bibliometrics study.* Expert Systems with Applications, 2018. **111**: p. 2-10.
2. Zheng, W., Y.-C.J. Wu, and L. Chen, *Business intelligence for patient-centeredness: a systematic review.* Telematics and Informatics, 2018. **35**(4): p. 665-676.
3. Chen, H., R.H. Chiang, and V.C. Storey, *Business intelligence and analytics: From big data to big impact.* MIS quarterly, 2012: p. 1165-1188.
4. Niu, Y., et al., *Organizational business intelligence and decision making using big data analytics.* Information Processing & Management, 2021. **58**(6): p. 102725.
5. Katarya, R., *Movie recommender system with metaheuristic artificial bee.* Neural Computing and Applications, 2018. **30**(6): p. 1983-1990.
6. Gupta, U. and N. Patil. *Recommender system based on hierarchical clustering algorithm chameleon*. in *2015 IEEE international advance computing conference (IACC)*. 2015. IEEE.
7. Kumar, M., et al., *A movie recommender system: Movrec.* International Journal of Computer Applications, 2015. **124**(3).
8. Ajesh, A., J. Nair, and P. Jijin. *A random forest approach for rating-based recommender system*. in *2016 International conference on advances in computing, communications and informatics (ICACCI)*. 2016. IEEE.
9. Frémal, S. and F. Lecron, *Weighting strategies for a recommender system using item clustering based on genres.* Expert Systems with Applications, 2017. **77**: p. 105-113.
10. Nilashi, M., et al., *A recommender system for tourism industry using cluster ensemble and prediction machine learning techniques.* Computers & industrial engineering, 2017. **109**: p. 357-368.
11. Soundarya, V., U. Kanimozhi, and D. Manjula, *Recommendation System for Criminal Behavioral Analysis on Social Network using Genetic Weighted K-Means Clustering.* J. Comput., 2017. **12**(3): p. 212-220.
12. Katarya, R. and O.P. Verma, *An effective web page recommender system with fuzzy c-mean clustering.* Multimedia Tools and Applications, 2017. **76**(20): p. 21481-21496.
13. Paradarami, T.K., N.D. Bastian, and J.L. Wightman, *A hybrid recommender system using artificial neural networks.* Expert Systems with Applications, 2017. **83**: p. 300-313.

14. Wasid, M. and R. Ali, *An improved recommender system based on multi-criteria clustering approach.* Procedia Computer Science, 2018. **131**: p. 93-101.
15. Zarzour, H., Z.A. Al-Sharif, and Y. Jararweh. *RecDNNing: a recommender system using deep neural network with user and item embeddings*. in *2019 10th International Conference on Information and Communication Systems (ICICS)*. 2019. IEEE.
16. Gabriel de Souza, P.M., D. Jannach, and A.M. Da Cunha, *Contextual hybrid session-based news recommendation with recurrent neural networks.* IEEE Access, 2019. **7**: p. 169185-169203.
17. Selvi, C. and E. Sivasankar, *A novel optimization algorithm for recommender system using modified fuzzy c-means clustering approach.* Soft Computing, 2019. **23**(6): p. 1901-1916.
18. Selvi, C. and E. Sivasankar, *A novel Adaptive Genetic Neural Network (AGNN) model for recommender systems using modified k-means clustering approach.* Multimedia Tools and Applications, 2019. **78**(11): p. 14303-14330.
19. Serrano, W., *Intelligent recommender system for big data applications based on the random neural network.* Big Data and Cognitive Computing, 2019. **3**(1): p. 15.
20. Cintia Ganesha Putri, D., J.-S. Leu, and P. Seda, *Design of an unsupervised machine learning-based movie recommender system.* Symmetry, 2020. **12**(2): p. 185.