Products And Movie Recommendation System For Social Networking Sites

Debajit Datta, T. M. Navamani, Rajvardhan Deshmukh

Abstract: Recommendation systems are an integral part of information filtering system in data science, that are widely used in order to identify the pattern a user would likely choose on the basis of the previous choices of the user as well as from studying the pattern in which others have chosen. For a fact, the recommendation can never be a cent percent correct at providing recommendations to the user but can be close enough to please them to a certain extent. Thus, the same is widely used in the industries these days to get higher profit and have a good hold in the market. The data scientists of every company design some algorithm that studies the information from the social network and clusters the data. There can be a single algorithm for classification like k-Means clustering or Hidden Markov model or can be done by bagging and boosting techniques. With this technique of displaying the movies or products into the profile of a particular customer, they not only increase their business but also enhances the customer experiences but there are several issues related to the standard techniques like the cold start problem, shrill attack, etc. thereby increasing the scope of research in this field. This work deals with both Collaborative Filtering and Content-Based Filtering to form a product and movie recommendation system for the social networking sites that shows the effectiveness of collaborative filtering and portrays the challenges faced by content-based filtering.

Index Terms: Collaborative Filtering, Content-Based Filtering, Classification, Clustering, Movie, Product, Recommendation.

1 INTRODUCTION

IN today's era, recommendation systems can be identified as an integral part of the information filtering system in the field of data science. These recommendation systems, these days, are widely used for mainly identifying the patterns or similarities amongst set of data which are not classified. These recommendation systems are knowingly designed in such a manner, that they can actually predict with significant amount of accuracy what a user would like to be recommended. This is done on the basis of their previous choices as well as after thoroughly studying the pattern in which other users who have shown interest in same thing or were under similar circumstances as the user. It is also used for feasibility checks, before establishment of a business or launching a new product. Even though these recommendation systems can provide highly accurate result of classifications, but still for a fact, they can rarely be a cent percent correct all the time at providing recommendations to the users. But an estimation can be made, which can be very close to please the users up to a certain extent. Thus, this fascinating idea of predicting human behaviour has inspired us in working with recommendation systems. This work deals with Collaborative Filtering to form a product and movie recommendation system for the social networking sites.

Social Networking sites are the new hotspots for this generation - people of all age group can be seen on these platforms. The online marketing is based on the survey and study taken from these platforms. Once the data is collected from various sources, several statistical methods and machine learning algorithms can be applied to understand the pattern and classify the data accordingly. With advancement in machine learning and data science, the efforts for creating an ideal system for recommending products or movies are increasing day by day. The marketing strategies include targeting of customers (for product-based systems) or viewers (for movie-based systems), so that the sales in the market as well as their profits reach a higher margin. The increase of efforts made is only due to the competition in the market. Movie and TV series streaming sites like Netflix have been putting their researches in order to target the exact viewer, or to grab attention of exact class or age-group of people; at the same time, to give them a good competition, organizations like Amazon Prime, Voot are also targeting people locally, for a greater impact on them. Parallelly, in products market, especially in India, Flipkart has been doing great in India, in targeting their customer based on the products they click on, or from the products they show interest to, or the ones the customers have mention in their posts or comments in some other social network [1][12]. All the data are read and stored in their database, they also have a lot of data coming from clickstream technology, where on clicks or halt-time or scroll speed are stored in order to estimate the interest on a particular product by user. These data are later clustered on various grounds, and the best product recommendations are given to the users [14][23]. They are not their own competition, companies like Amazon, Myntra and many others have also developed their own algorithms and techniques in order to focus on their customers. Widely, the categorization of these recommendation systems is into two major types: Collaborative Filtering Approach and Content-Based Approach. For collaborative filtering, the final information is, later, computed using the data consisting of user item rating matrix for prediction purposes. Additionally, content-based approach gives recommendation to the user based on the history of the user, by studying their social network interactions, rather than the similar items or users [4]. None of these algorithms can be considered as ideal, because none of these are ever cent percent accurate, when it comes to testing the model. Out of these two approaches Collaborative based item or user filtering approaches are the most popular one because of their efficiency. The reason behind popularity of collaborative filtering over content-based approach is mainly due to the fact that the latter one considers the history of only the user, and recommends on the basis of it, but it is quite possible that the user might have lost interest in the genre of movie or might not be into the product anymore [2][20] whereas, on the other hand with collaborative filtering, one

- *Debajit Datta is currently pursuing bachelors degree program in computer science and engineering in Vellore Institute of Technology, Vellore, India. E-mail: debajit.datta2000@gmail.com*
- *T. M. Navamani is an associate professor of school of computer science and engineering (SCOPE) in Vellore Institute of Technology, Vellore, India. E-mail: navamani.tm@vit.ac.in*
- *Rajvardhan Deshmukh is currently pursuing bachelors degree program in computer science and engineering in Vellore Institute of Technology, Vellore, India. E-mail: rajvardhan1999@gmail.com*

gets recommendation based on the similarity, so one might get recommendation of other latest movies when they show interest in one of them, similarly, say, someone shows interest in buying shaving razor, then it is likely to recommend them shaving gel or aftershave lotion, irrespective of their past searches. There are many other kinds of methodologies/techniques that have been introduced in order to implement this Recommendation System deals with various concepts like data mining, clustering of data, and implementation of Bayesian networks. Out of these techniques, Bayesian networks works most effectively but the main disadvantage of the Bayesian networks is that they cannot be applied for the systems in which the information and data from which the system is extracting the preferences is frequently changing. In addition to these classes of techniques, there exists another class of recommender system combining two or more types of recommendation techniques into one system, termed as a hybrid. It combines the relevant attributes, which will be good for consideration in developing a recommender system, thereby reducing the anomalies. Even though these are having huge popularity, efficiency and wide range of applicability and acceptability still faces several problems including cold start problems, data sparsity and shriller attacks etc. To solve the problem of sparse user-item matrix various techniques like Singular Value Decomposition and models like Bayesian classifiers, matrix factorization and genetic algorithms are used. Various clustering techniques like a Particle Swarm optimization, an Ant Colony optimization and k-means are used these days to improve the quality of predictions [27][28]. This method provides solutions to remove the cold start problem, but are expensive in terms of computations and complexity. This research provides a technique based on Collaborative Filtering to improve the available recommendation systems that are available and overcome the above-mentioned issues. This work specifically implements two independent modules: product recommendation module and movie recommendation. It proposes to resolve most of the issues like cold start problems, data sparsity and shriller attacks. To be specific, this work is based on the recommendations that are obtained by influence and recommendation – on products and movies, that are commonly bought together for movie recommendation system and pure mathematical collaborative algorithm for the product recommendation system. This work also tries to demonstrate how content-based filtering can be explicitly adopted for recommending to users – by mimicking it within a movie recommendation system prototype. This work is divided into two broad modules which has several sections – (i) a Python oriented collaborative-filtering based product recommendation system which is implemented on two different Kaggle-based Amazon Product datasets: Office Product and Digital Music dataset; (ii) a MATLAB oriented collaborative-filtering based movie recommendation system which is implemented on MovieLens dataset of movie rating by GroupLens Research; (iii) a Python oriented movie recommendation system implemented with content-based filtering on IMDB movies dataset by Data-World. The organization of the work is structured as follows: Section 2 of this work discusses the literature studies that have been referred for developing the system; Section 3 gives an insight to the proposed system; Section 4 is devoted to the methodology and implementation of this work; Section 5 summarizes the observations made through this work and their outcomes; Section 6 concludes the work and proposes the further studies that can be carried out in future based on this work.

2 Related Work

For accomplishment of this work, several research works and journals have been referred in order to come up with better ideas and improvements of the existing systems. With advancement in machine learning and data science, the efforts for creating an ideal system for recommending products or movies are increasing day by day. All the data are read and stored in their database, they also have a lot of data coming from clickstream technology, where on clicks or halt-time or scroll speed are stored in order to estimate the interest on a particular product by user. The work of Nandagawali and Patil [1] provides a solution for the various recommender systems by using collaborative filtering algorithms with the community-based user domain model. The main purpose is to satisfy the customer's product needs by providing them recommendation based on products. It deals with the Amazon dataset. Better hardware configuration may result in better output. The work of Jatinder et al., [2] deals with the review of several techniques that are used by a system for recommending electronic products. They have also worked with the Amazon dataset. Time complexity of the whole model is pretty high, making it less convenient. In the work of Eklaspur and Pashupatimath, [3] a formal representation of social network is present where text mining is taken as a perspective. A framework is proposed that can recommend friend using an efficient Algorithm. Their work is based on Facebook data. The notable points in their work can be that the accuracy is high, and works fine with any size of data. Analysis of the classes and its classification is also highly précised. But it lacks to meet the standard when it comes to time of execution. Haruna, et al., [4] in their work, have proposed a research paper recommender system that transforms all the recommending papers into a paper-citation relations matrix. The key points that makes their model better can be faster results. Their algorithm which is Brute Force based, but fails to meet the compatibility in real time application as it will be inefficient for very large dataset, as the number of rows to column ratio will increase abnormally. In the research work carried out by Kumar et al., [5] they have proposed a movie recommendation system named MOVREC. The system sorts the ratings by implementing K-means clustering algorithm. The pros of the system include that the time taken is less, and almost every time the result of the classification of data is accurate [21][25]. But the cons can be that creating classes amongst closely related data which may belong to the same cluster can be redundant and misleading. This work has also referred to the paper proposed by Cui [6] the author has designed and implemented a movie recommendation system prototype combined with the actual requirements of movie recommendation by implementation of K-Nearest Neighbours (KNN) algorithm, which is supported by a collaborative filtering algorithm. The advantage of the system is that the algorithm is faster compared to other neural networks like the Convolutional Neural Network (CNN) or the Recurrent Neural Network (RNN) or the Artificial Neural Network (ANN). Owing to the various demerits of pure content-based and pure Collaborative Filtering based systems, the work carried out by researchers [7][26] proposes a hybrid recommender system in order to use a content-based predictor with collaborative filtering to fill the user-rating matrix

that is sparsely distributed. The dataset, made by the help of web-crawlers, consists of a user-rating matrix. Several works [23][24] should be given credit for efficient algorithm and its reliability. In the research carried out by Sharma [8], authors proposed a movie recommender system based on new user similarity metric and opinion mining. The system abstracts aspect-based detailed ratings from reviews and also recommend reviews to users. Their work is inspired by the k-means clustering algorithm for unsupervised learning. According to the work of Chen et al. [9], they have incorporated Singular Value Decomposition (SVD) in user based collaborative filtering technique for movie recommendation system. Although, they have got good accuracy with their model, but additional user or item features can be applied along with a larger dataset for better preparation of model. Lin et al. [10] have created a movie recommendation system which is based on collaborative filtering and neural network. Their approach is innovative and have also achieved low error. Several researches [11][19], deal with implementation of Markovian Factorization for movie recommendation. They have obtained high accuracy based on several parameters. Some works [12][20][22] have used rating variance for creating tag-based product recommendation system. Consideration of only rating, may not be sufficient for product recommendation, rather class and type of item must also be taken into account. The work of Reddy, et al. [13] deals with genre correlation-oriented content-based movie recommendation system, that only focusses on the genre of a movie. On the contrary, for any person, choice might not be completely based on one genre. Several papers [14][29] deal with several trends in content-based recommenders as well as the possible subdivisions in data related and algorithmic trends in content-based recommenders. Some research works that are carried out [15][30], also deal with the comparison of several recommendation systems. Their paper has inspired this work to focus mainly on the Collaborative Filtering, because Content Based Filtering can be expensive when it comes to a large number of users. Every research work that has been referred shows the hard work of their associated authors. Despite of being informative, some scope of improvement was available in each of the research works. This work has tried to overcome those shortcomings, by coming up with implementation of both Collaborative Filtering and Content Based Filtering methods. In the existing models, less parameters are focused while providing a recommendation, but this system works with multiple parameters while recommending users. Also, this system provides better accuracy and yields better F-score.

3 Proposed Work

3.1 Review Stage

Recommendation systems can be obtained by several machine learning algorithms like Decision Tree, K-means Clustering, etc. But, in order to generalize the categorization of the recommendation systems, it can be divided into two major types: collaborative filtering approach and content-based approach. Although they are different algorithms, but they share same strategy of using the similarities amongst the users or items. For collaborative filtering, the final information is, later, computed using the data consisting of user item rating matrix for prediction purposes. Additionally, content-based approach gives recommendation to the user based on the history of the user, by studying their social network interactions, rather than the similar items or users. None of these algorithms can be considered as ideal, because none of these are ever cent percent accurate, when it comes to testing the model. Collaborative based item or user filtering approaches are the most popular one because of their efficiency. Tapestry was the first one to use this collaborative filtering techniques to implement recommender systems. In that system the preferences provided by the users are first extracted from the ratings that are provided by the user explicitly or implicitly. After this a large number of methodologies has been introduced in order to provide personalized recommendations to the user. Ringo video Recommender system is a web-based application that generates recommendations to the user on movies, Videos and music and many more. Group lens also developed a recommender system using item based collaborative filtering approach that provides recommendations for news, Movies etc. In this work, more concentration is towards collaborative filtering approach of recommendations. Collaborative filtering is chosen over content-based approach mainly because of the fact that the latter one, that is, content-based filtering is based on only the history of the user, and recommends on the basis of it, and fails to consider the possibility that the user might have lost interest in, say, the genre of movie or might not be into the product anymore. Whereas, with collaborative filtering, one gets recommendation based on the similarity, so one might get recommendation of other latest movies when they show interest in one of them, similarly, say, someone shows interest in buying shampoo, then it is likely to recommend them conditioner, as general public buys conditioner with shampoo.

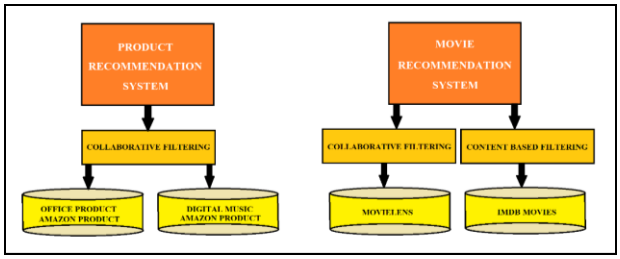

***Fig. 1**. System Architecture for (left) Product Recommendation and (right) Movie Recommendation systems.*

This system is broadly divided into two major categories, namely, the product recommendation system and the movie recommendation system as shown in Fig. 1. The product recommendation system is basically used for suggesting products based on classification of dataset that has products in it. The movie recommendation system works with suggestion of movies with respect to user feedback and their comments on social networking sites. The work is aimed to re-create scenarios that can be dealt in the real life, in order to present the systems for industrial use, or future integration with other applications or social networking sites like Facebook, Twitter, Google Plus, LinkedIn, Snapchat, Instagram, Redditt, Tumblr, BlogSpot, Medium, etc. The modules in this work, are independent of each other, and have different functionalities. In a wide scale, product recommendation is used by big flourishing companies like Flipkart, Amazon, Snapdeal, OLX, H&M, Alibaba, Big Basket, etc. They target customers on the basis of the products they

show interest in, rather than a pre-classified data, hence, the approach is not a supervised learning algorithm but is unsupervised, because it forms clusters according to the similarity ratios and other parameters, rather than the classes they belong to. The recommendations are generated only through trustworthy users. These recommender systems are generally applied for a wide range of problem domains these include media, entertainment, etc. For someone who wants to implement this recommender system first need to understand the end user taste and the preferences of him. The recommender system implemented in this work, should resemble the taste of the user and his requirements and must be suitable for the problem domain. It should also be able to integrate the preferences or the feedback that are provided by the user into a single unit data source such that the recommendations that are calculated, are going to represent the entire range of information provided by the user. This work uses a methodology that has been implemented to overcome a few of the problems like cold start problem, data sparsity and shriller attacks, by implementing adjacent cosine-based similarity for computation of similarities between two concepts and selection of neighbours and then use them for the prediction ratings. This system would later recommend items for the user. Advantages of this system include: improved prediction accuracy compared to other techniques like content-based filtering; it can even work well when there is a sparse training dataset, like the movie dataset that has been used in this work; it reduces the number of big error predictions.

3.1 Product Recommendation

This module of the work deals with recommending users the products based on various parameters inferred from the available dataset. The selection of parameters is a challenging task, since, based on the attribute, it might result in underfitted model or overfitted model. Models which are either underfitted or overfitted, gives poor result in accuracy on testing dataset, and thus are not preferred. This work implements the product recommendation based on collaborative filtering and the recommendation is highly focused on the influence of other users as well as the products itself.

3.1.1 Recommendation Based on Influence

This type of recommendation will be based on how the particular node is influenced by the other nodes. In the case of product recommendation, this method will allow the user to see a list of recommended items that he or she can buy along with the item that he or she has already chosen. Suppose item X is bought together with item Y. If the item X is bought with several other items as well, then we say that X and Y don't influence each other whereas, if the item X is mostly bought along with item Y then we say that item X and Y influence each other. This is called 'influence scoring'.

3.1.2 Recommendation by Commonly Bought Products

This method is like the transitive property. If X is bought together with Y and most of the people buying Y have also bought Z along with it, then item Z will also be recommended when any user is buying item X. Also, if the item Z is bought commonly then its chances of being recommended increases. This system performs collaborative filtering by considering the recommendations based on both, influence and products that are widely bought together.

3.2 Movie Recommendation

This module of work uses both, collaborative filtering technique and content-based filtering technique for implementation of movie recommendation system. Collaborative-filtering techniques deal with data of the particular user along with the data of other users. On the contrary, content-based filtering techniques deal with the data of the user alone. The collaborative filtering technique is very simple where the requirement is just to find the neighbors that share similar interests with that of the user and then recommend the user the movies that the neighbors, having similar taste, like. The basic idea is to recommend similar items (movies in this case) to the similar end-users of the recommendation system based on the history of ratings by the user for the items [26][28]. This collaborative filtering technique is divided into the following steps – Collaborative filtering learning algorithm: The collaborative filtering algorithm is dependent on a few parameters based on the movie and the rating by users. With these rating on some movies, by some user, the system will begin by finding out those parameters that affect this prediction and find out the best fit movie. For the ease of calculating, a cost function is set up to unroll these parameters into single vector params; Calculating collaborative filtering: The collaborative filtering calculations are carried out using the cost functions and gradients of the values that are obtained [29]; Regularized cost function: After the calculation of the collaborative filtering parameters, the calculated measures of cost function and the gradients are regularized to further work with it. The content-based filtering technique is divided into the following steps – identifying the actors, genres, directors and plots of the movies that the user has watched; apply content-based filtering algorithm: to suggest movies based on calculations.

4 Implementation Detail

4.1 Software and Hardware Requirement

The system has been developed on a windows 10 machine having i7 processor with 16 GB RAM. The software required for successful implementation were Jupyter Notebook and GNU Octave for executing MATLAB codes. The python libraries used for this system includes numpy, pandas, nltk, sklearn, math, scipy, network and matplotlib.

4.2 Dataset Implemented

For product recommendation system, Amazon Product datasets: Office Product and Digital Music dataset are used which have been taken from Kaggle, an online platform for data science enthusiasts. There are 1,243,186 reviews for Office Products while there are 836,006 reviews for Digital Music. The number of products in Office Products is 134,838 while that for Digital Music is 279,899 products. For movie recommendation system with collaborative filtering, MovieLens dataset is used which is provided by GroupLens. It consists of 1600 distinct movies along with 943 distinct users. The movie recommendation system is implemented with content-based filtering on Data-World IMDB movies dataset consisting of 5043 distinct movies.

4.3 Methodologies Used

The two modules of this work are distributed as Product Recommendation system and Movie Recommendation system. Whilst major focus has been given to collaborative

filtering, this work, also demonstrates how all of the recommendations can be based on several factors associated with a user, with incorporation of content-based filtering technique concepts. For product recommendation, this work uses two different classes of products namely, digital music and office products. These data are worked with collaborative filtering to show the accuracy of the models. Corresponding graphs have also been plotted for proper visualization with plots like dendogram clustering and scatterplot. Python 3 has been used to implement the same in Jupyter notebook. Jupyter notebook provides Python 3 notebook for execution of codes independently within the notebooks provided, along with options to document the same. Both of the datasets used are Amazon Product dataset are taken from the Kaggle platform. Kaggle platform is an open platform that provides datasets as well as it provides online python notebooks for execution of codes. This Kaggle platform is used by most data science and machine learning enthusiasts. Similarly, movie recommendation has been implemented and visualization has been done with proper plots. Movie ratings from IMBD and MovieLens dataset are used to implement this part of the work. The accuracy measures have also been taken in order to find out the total accuracy of the model as well as validate it. The section of this module dealing with collaborative-filtering for recommendation uses MovieLens dataset and has been implemented using MATLAB. MovieLens dataset is taken from the organization named GroupLens. The other section, that involves content-based filtering for recommendation, uses IMDB movies data, and has been implemented using Python 3 in Jupyter Notebook. The dataset that includes IMDB movies, consists of 5043 distinct movies, and it has been taken from Data-World platform. The clusters that are found are also plotted with proper graphical representation for having a clear visualization of how the clusters are formed. Along with that, the accuracy of the collaborative filtering systems has also been shown. The demonstration of growth of nodes and connections within the graph associated with content-based filtering is also represented within this piece of work. Finally, the conclusion has been drawn on the basis of the observations. Implementation of collaborative filtering learning algorithm is obtained by implementing the cost function (without regularization). The collaborative filtering algorithm for executing a recommendation uses n-dimensional parameter vectors x(1), ..., x(nm) which are the features and θ(1), ..., θ(nu). The features can be movie based or product based in case of this system. The model is trained to predict the rating for movie or product i by user j as y(i,j) = (θ(j)) T x(i). A dataset which comprises of a set of ratings obtained by various users, is expected to produce the best fit. A best fit model is the one that works fine with both training and testing dataset. The corresponding equation [16] is represented in (1). The parameters of the function are X and Theta. For implementation of a minimizer like fmincg, the cost function needs to be set up to unroll the parameters into single vector parameters.

$$J\left(x^{(1)}, \dots, x^{(n_m)}, \theta^{(1)}, \dots, \theta^{(n_u)}\right) = \frac{1}{2}\sum_{(i,j):r(i,j)=1}\left(\left(\theta^{(j)}\right)^T x^{(i)} - y^{(i,j)}\right)^2 \quad (1)$$

The collaborative filtering cost function is given by (1), which can be used to calculate the cost function in a separate function. The overall cost accumulates the cost for user j and movie or product i only if R(i,j) = 1. After this, the gradient is implemented using (2) for products and movies and (3) for Theta [16]. The computation generates the variables X gradient and Theta gradient as displayed in the two equations. While calculating the gradients of both X and Theta, it should also be noted that the gradient X should be a matrix of the same size as that of X and at the same time, the Theta gradient should be a matrix of the same size as that of Theta.

$$\frac{\partial J}{\partial x_k^{(i)}} = \sum_{j:r(i,j)=1}\left(\left(\theta^{(j)}\right)^T x^{(i)} - y^{(i,j)}\right)\theta_k^{(j)} \quad (2)$$

$$\frac{\partial J}{\partial \theta_k^{(j)}} = \sum_{i:r(i,j)=1}\left(\left(\theta^{(j)}\right)^T x^{(i)} - y^{(i,j)}\right)x_k^{(j)} \quad (3)$$

The function returns the gradient for both sets of variables by unrolling them into a single vector. After computing the gradient from (2) and (3), a gradient check is done to numerically check the implementation of the gradients. It helps in find that the analytical and numerical gradients match up closely signifying correct gradients. The second part of second module of this system deals with content-based filtering on movie recommendation. For carrying out the content-based filtering, the dot products of the vectors are produced separately in order to find the similarity between the movie items in order to recommend the users the particular movies [17]. Suppose, if u is considered to be the vector corresponding to users and v to the training dataset containing movies, (4) and (5) produces the dot products of them.

$$u \cdot v = [u_1\ u_2\ \dots\ u_n] \cdot [v_1\ v_2\ \dots\ v_n]^T \quad (4)$$

$$u \cdot v = u_1v_1 + u_2v_2 + \cdots + u_nv_n = \sum_{i=1}^{n} u_iv_i \quad (5)$$

The similarity is found by taking the cosine similarity amongst the selected two vectors [17] as shown in (6), in the case of this project, it is the movie vector.

$$similarity = cos(\theta) = \frac{u \cdot v}{\|u\|\,\|v\|} = \frac{\sum_{i=1}^{n} u_iv_i}{\sqrt{\sum_{i=1}^{n} u_i^2}\sqrt{\sum_{i=1}^{n} v_i^2}} \quad (6)$$

After the similarity is calculated using (6), the movie that is associated with vector having highest cosine similarity value is chosen to be the most recommended movie for the user.

Finally, in order to calculate the accuracy of the system, the recommendations are compared with that of original recommendation. The true positive, true negative, false positive and false negative values are calculated and corresponding precision and recall are calculated and then the F-score is also calculated with the help of calculated precision and recall.

5 Results and Analysis

This section of the paper deals with the various observations that are made after implementing the system. The accuracy of the system has been found out by considering several parameters of the classifications like, director for movies and rating for products. The graphs have also been plotted with the help of several libraries like matplotlib, networkx, etc. This work includes several graphs like dendrograms for visualization of hierarchical clustering, scatter plot and network graph plot for visualizing how graphs increase with time and clusters are formed. The different plots used in this work provides better visualization and better understanding of the clustering done by the system.

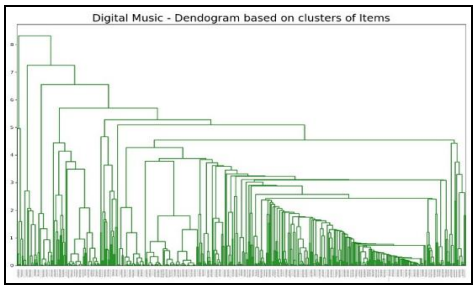

***Fig. 2**. Dendrogram Clusters for Digital Music item cluster for five reviewers.*

The visualization of dendrogram clusters for Digital Music are visualized in Fig. 2 for item clusters for five randomly chosen reviewer and in Fig. 3 for reviewer clusters for five randomly chosen items. The dendrogram clusters are a part of hierarchical clustering, that are clustered so that recommendations can be given on the basis of the clusters. The hierarchical clustering helps to understand how the clustering are done in various stages.

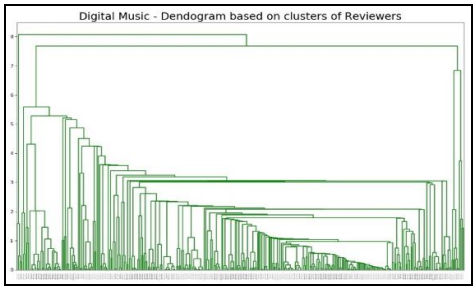

***Fig. 3**. Dendrogram Clusters for Digital Music reviewer cluster for five items.*

The clusters in Fig. (2) and Fig. (3) that are formed far from the base line are weakly clustered, whereas, the others are strongly clustered. Thus, while recommending a digital music, the strongly clustered ones will be selected.

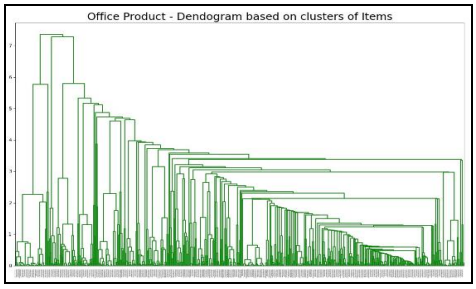

***Fig. 4**. Dendrogram Clusters for Office Product item cluster for five reviewers.*

The visualization of dendrogram clusters for Office Product are visualized in Fig. 4 for item clusters for five randomly chosen reviewer and in Fig. 5 for reviewer clusters for five randomly chosen items. These clusters that are formed due to hierarchical clustering, helps in understanding the underlying clusters that can be formed conceptually amongst the different reviewers and different items.

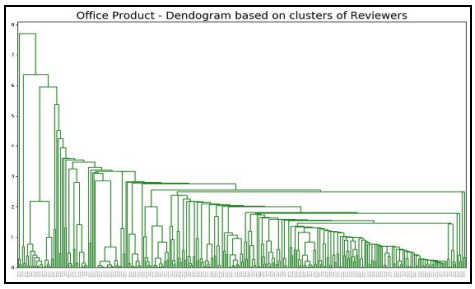

***Fig. 5**. Dendrogram Clusters for Office Product reviewer cluster for five items.*

Although the visualization is only for few items and a few reviewers, in reality all items and reviewers are considered when the clusters are formed. This enables in selecting the office products within a created cluster, in order to recommend it to the users belonging to the same cluster.

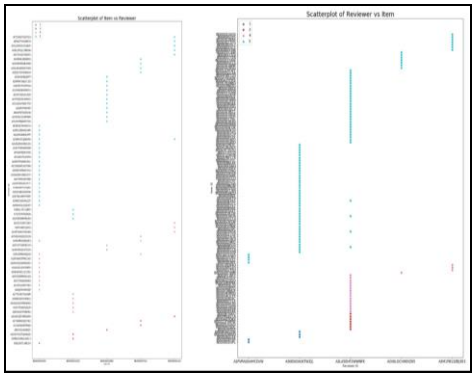

***Fig.** 6. Scatterplot for Digital Music (left) items vs. reviewers (right) reviewers vs. items.*

The scatterplot visualization graph for Digital Music products are shown in Fig. 6, left image, for the items versus the reviewers which consists of five randomly selected items and their associated reviewers who have reviewed those items and Fig. 6, right image, for the reviewers versus the items which consists of five randomly selected reviewers and the items they have reviewed. This scatter plot helps to understand the like-minded customers, thus will help recommend them digital music accordingly. The scatterplot visualization graph for Office Product items are shown in Fig. 7, left image, for the items versus the reviewers which consists of five randomly selected items and their associated reviewers who have reviewed those items and Fig. 7, right image, for the reviewers versus the items which consists of five randomly selected reviewers and the items they have reviewed. The plot helps understand the reviewers that share equal interest towards the office products, and thus help them recommend the office products accordingly.

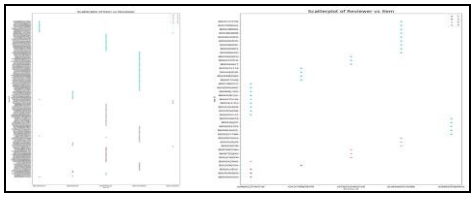

***Fig. 7**. Scatterplot for Digital Music (left) items vs. reviewers (right) reviewers vs. items.*

For the scattered plots, the reviews are plotted and the color is chosen by the rating values like in Fig. 6 and Fig. 7. For a given item, be it Office Product or Digital Music, the number of reviewers is in hundreds or thousands and are represented

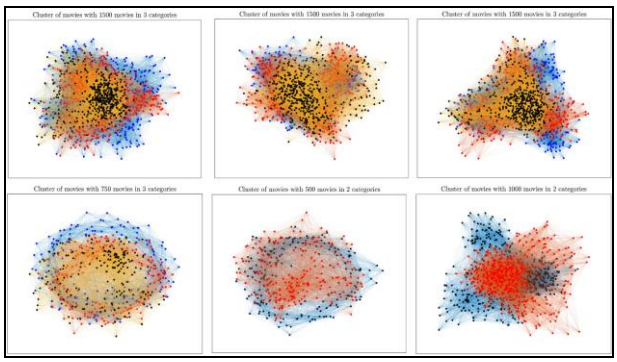

Fig. 8*. Growth of clusters of movies with collaborative filtering (top, left) 1500 movies belonging to 3 categories (top, middle) 1500 movies belonging to 3 categories (top, right) 1500 movies belonging to 3 categories (bottom, left) 750 movies belonging to 3 categories (bottom, middle) 500 movies belonging to 2 categories (bottom, right) 1000 movies belonging to 2 categories.*

in the y-axis and for selected reviewer, the number of products is also pretty high. This plot gives an idea of the vastness of comparison that is taken into consideration for implementing a collaborative filtering. The movie recommendation system is visualized through both collaborative filtering and content-based filtering. The growth of clusters, the formation of new clusters and ending of a cluster with increase in the number of nodes and time are plotted through MATLAB and networkx on python. The movies of a particular genre are considered as nodes and the edges between them show how they are clustered. For collaborative filtering, MATLAB is used to demonstrate the growth of graph as shown in Fig. 8 and for content-based filtering, as shown in Fig. 9, networkx library of Python has been used. The MATLAB file “WattsStrogatz.m” is used for creating the several graphs referred to as small world graphs in order to show the clustering within the movies.

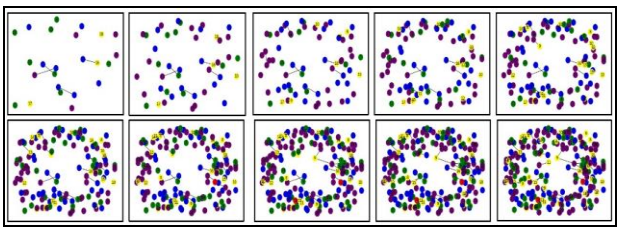

Fig. 9. Growth of clusters of movies with content-based filtering.

The growth for a collaborative-filtering implemented recommendation system can be seen in Fig. 6, where the clusters continuously keeps changing time to time irrespective of number of movies or nodes and genres or categories of movies. This change is not constant, and changes with time. The growth of clusters of movies can be seen in Fig. 9 where all the movies of different genres are clustered together for recommendation purpose. The different colors of nodes denote the different categories of movies that are available within the dataset. Based on the clusters that are dynamically formed, the recommendations are given to a person. Although

TABLE 1

TABULATION OF ACCURACY METRICS FOR THE SYSTEM

Metric	Digital Music	Office Product	Movie
True Positive	84780	82221	80613
True Negative	3722	1305	5090
False Positive	4051	2329	12391
False Negative	529	782	1917
Precision	0.9544	0.9724	0.8668
Recall	0.9938	0.9905	0.9768
F-Score	0.9736	0.9814	0.9185

this graph is only based on few clusters, in reality, there are many more of nodes available. Content-based filtering works to recommend a particular user based on their previous choices. The movie recommendation system takes name of movie as an input, and based on the input, provides with the recommended movies for that particular input movie. For example, if the input movie is ‘Fargo’, the output is ['No Country for Old Men', 'The Departed', 'Rope', 'The Godfather', 'Reservoir Dogs', 'The Godfather: Part II', 'On the Waterfront', 'Goodfellas', 'Arsenic and Old Lace', 'The Big Lebowski'] and when the input movie is ‘Finding Nemo’, the output is ['Toy Story 3', 'WALL·E', 'Monsters, Inc.', 'Up', 'Toy Story', 'The Nightmare Before Christmas', 'Aladdin', 'Zootopia', 'Song of the Sea', 'Inside Out'] or when the input is 'The Wolf of Wall Street', the recommended movies are ['Goodfellas', 'The Godfather: Part II', 'The Departed', 'The Godfather', 'Catch Me If You Can', 'Touch of Evil', 'Cool Hand Luke', 'Baby Driver', 'Sin City', 'A Clockwork Orange']. This system, that has been developed, provides great accuracy according to the various globally accepted metrics like Precision, Recall and F-Scores. The Precision metric is calculated as ratio of True Positive to the sum of True Positive and False Positive whereas, the Recall metric is calculated as ratio of True Positive to the sum of True Positive and False Negative. The F-Score metric is dependent on Precision and Recall. The F-Score is calculated by taking the ratio of product of Precision and Recall to the average of Precision and Recall [18], which basically the harmonic mean of the two parameters termed as Recall and Precision. For the Digital Music recommendation under Product recommendation, the achieved precision is 0.9544, the achieved recall is 0.9938 and the achieved F-score is 0.9737. For the Office Product, the metric obtained are 0.9724 for precision and 0.9905 for recall and F-score is 0.9814. For the collaborative-filtering based movie recommendation system, the precision achieved is 0.8668 and recall achieved is 0.9768, thus the calculated F-score is 0.9185. Further information regarding the metric is tabulated in Table 1.

The overall accuracy of the system is pretty high for both modules, namely, product recommendation system and movie recommendation system. The recommendations are also appropriate when compared with the actual dataset. The graphs generated help in understanding how the clusters are formed and how the clusters grow with time. The true positives are those instances which are true in the actual dataset and are also classified as true after inclusion of the model, similarly the true negative instances are those which are false for both cases. The overall accuracy of the system is pretty high for both modules, namely, product recommendation system and movie recommendation system. The recommendations are also appropriate when compared with the actual dataset. The graphs generated help in understanding how the clusters are formed and how the clusters grow with time. The true positives are those instances which are true in the actual dataset and

are also classified as true after inclusion of the model, similarly the true negative instances are those which are false for both cases. The false positives are those instances which are false in actual dataset but calculated as true in the output, and similarly, false negatives are vise-versa. The precision and recall of this system are very high and also the associated F-Score.

6 Conclusion

The system that has been created serves with optimal and accurate results compared to that of the associated dataset. The F-score which is widely accepted metric, gives really high value close to that of an ideal system. The associated graphs show how the clusters are formed alongside helps in visualization of how the clusters are formed in a network in real life. The networks can be products network or can be movies network, but the clusters are dynamic and changes with time. The same technology can be incorporated within the social networks to target the people, in order to boost up the overall sale of the product-based or movie-streaming based companies. The connection between any two nodes is not permanent. The number of attributes that are considered for training this system are optimal because too many attributes can result in overfitting, but too less attributes can tend to underfitting. But the attributes chosen must be oriented with the output, so that, it results in proper fitting of the models with better training. It is also observed that compared to content-based filtering, collaborative filtering is easier to achieve because the former one deals with history of an individual along with the associations with other users. It is thus concluded that the implementation of content-based filtering is more challenging in real life compared to that of collaborative filtering. Though, with technologies like big data and clickstream, the incorporation of them can be useful in achieving the real-life dataset, and high-performance computing can also be included for execution of both content based and collaborative filtering. This work can be enhanced in future by including technologies like big data and high-performance computing. The dataset can be increased including real-time data as well. Other new clustering algorithms can also be added to further increase the overall accuracy of the system. In future, these clustering algorithms can also be incorporated with neural networks to train the model in a better way.

References

[1] Nandagawali, Priyanka A., and Jaikumar M. Patil. "Community based recommendation system based on products." 2014 International Conference on Power, Automation and Communication (INPAC).

[2] Kaur, Jatinder, Rajeev Kumar Bedi, and S. K. Gupta. "Product Recommendation Systems a Comprehensive Review." (2018).

[3] Eklaspur, Namrata M., and Anand S. Pashupatimath. "A friend recommender system for social networks by life style extraction using probabilistic method-friendtome." International Journal of Computer Science Trends and Technology (IJCST) 3.3 (2015).

[4] Haruna, Khalid, et al. "A collaborative approach for research paper recommender system." PloS one 12.10 (2017): e0184516.

[5] Kumar, Manoj, et al. "A movie recommender system: Movrec." International Journal of Computer Applications 124.3 (2015).

[6] Cui, Bei-Bei. "Design and implementation of movie recommendation system based on Knn collaborative filtering algorithm." ITM web of conferences. Vol. 12. EDP Sciences, 2017.

[7] Hande, Rupali, et al. "MOVIEMENDER-A movie recommender system." International journal of engineering sciences & research technology (IJESRT) 5.11 (2016): 686.

[8] Sharma, Pooja Mr Bhupender. "Movie Recommendation System: A Review Report." Journal for Research| Volume 4.01 (2018).

[9] Chen, Vito Xituo, and Tiffany Y. Tang. "Incorporating singular value decomposition in user-based collaborative filtering technique for a movie recommendation system: A comparative study." Proceedings of the 2019 the International Conference on Pattern Recognition and Artificial Intelligence. 2019.

[10] Lin, Chu-Hsing, and Hsuan Chi. "A novel movie recommendation system based on collaborative filtering and neural networks." International Conference on Advanced Information Networking and Applications. Springer, Cham, 2019.

[11] Zhang, Richong, and Yongyi Mao. "Movie Recommendation via Markovian Factorization of Matrix Processes." IEEE Access 7 (2019): 13189-13199.

[12] Tewari, Anand Shanker, and Ashu Mainwal. "Tag based product recommendation system using rating variance." Proceedings of 2nd International Conference on Advanced Computing and Software Engineering (ICACSE). 2019.

[13] Reddy, S. R. S., et al. "Content-based movie recommendation system using genre correlation." Smart Intelligent Computing and Applications. Springer, Singapore, 2019. 391-397.

[14] Lops, Pasquale, et al. "Trends in content-based recommendation." User Modeling and User-Adapted Interaction 29.2 (2019): 239-249.

[15] Dhawan, Sanjeev. "Comparision of Recommendation System Approaches." 2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon). IEEE, 2019.

[16] "Recommender Systems (Implementing In Octave) - UPSCFEVER." Upscfever.com. Web. 19 June 2020. <https://upscfever.com/upsc-fever/en/data/en-exercises-25.html>.

[17] "9.5.2. The Cosine Similarity Algorithm - 9.5. Similarity Algorithms." Neo4j.com. Web. 23 June 2020. <https://neo4j.com/docs/graph-algorithms/current/labs-algorithms/cosine/>.

[18] Brownlee, Jason. "Classification Accuracy Is Not Enough: More Performance Measures You Can Use." Machine Learning Mastery. N.p., 2019. Web. 10 June 2020. <https://machinelearningmastery.com /classification-accuracy-is-not-enough-more-performance-measures-you-can-use/>.

[19] Natarajan, Senthilselvan, et al. "Resolving data sparsity and cold start problem in collaborative filtering recommender system using linked open data." Expert Systems with Applications 149 (2020): 113248.

[20] Datta, Debajit, et al. "Comparison of Performance of Parallel Computation of CPU Cores on CNN model." 2020 International Conference on Emerging Trends in

Information Technology and Engineering (ic-ETITE). IEEE, 2020.
[21] Aljunid, Mohammed Fadhel, and Manjaiah Dh. "An Efficient Deep Learning Approach for Collaborative Filtering Recommender System." Procedia Computer Science 171 (2020): 829-836.
[22] Alhijawi, Bushra. "Improving Collaborative Filtering Recommender System Results using Optimization Technique." Proceedings of the 2019 3rd International Conference on Advances in Artificial Intelligence. 2019.
[23] Datta, Debajit, and Dheeba J. "Exploration of Various Attacks and Security Measures Related to the Internet of Things." International Journal of Recent Technology and Engineering (IJRTE) 9.2 (2020): 175-184.
[24] Mohammadpour, Touraj, et al. "Efficient clustering in collaborative filtering recommender system: Hybrid method based on genetic algorithm and gravitational emulation local search algorithm." Genomics 111.6 (2019): 1902-1912.
[25] Mokarrama, Miftahul Jannat, Sumi Khatun, and Mohammad Shamsul Arefin. "A content-based recommender system for choosing universities." Turkish Journal of Electrical Engineering & Computer Sciences 28.4 (2020): 2128-2142.
[26] Deldjoo, Yashar, Markus Schedl, and Mehdi Elahi. "Movie genome recommender: A novel recommender system based on multimedia content." 2019 International Conference on Content-Based Multimedia Indexing (CBMI). IEEE, 2019.
[27] Anand, Poonam Bhatia, and Rajender Nath. "Content-Based Recommender Systems." Recommender System with Machine Learning and Artificial Intelligence: Practical Tools and Applications in Medical, Agricultural and Other Industries (2020): 167.
[28] Loboda, Olga, et al. "Content-based Recommender Systems for Heritage: Developing a Personalised Museum Tour." Proceedings DSRS-Turing'19. London, 21-22nd Nov, 2019 (2019).
[29] Datta, Debajit et al. "Neural Machine Translation Using Recurrent Neural Network." International Journal of Engineering and Advanced Technology (IJEAT) 9.4 (2020): 1395-1400.
[30] Cami, Bagher Rahimpour, Hamid Hassanpour, and Hoda Mashayekhi. "User preferences modeling using dirichlet process mixture model for a content-based recommender system." Knowledge-Based Systems 163 (2019): 644-655.