Framework of Web Recommendation System For Browsing Behaviour Prediction

Sowmya K Menon, Varghese Paul

***Abstract:**Social networking is an important aspect in our day today life. In this era our main aim is to develop systems recommending good types of websites to users. As the changes in social networking is increasing day by day ,there should be the need for a system recommending the websites the user is having great interest. This paper focuses on recommending systems for browsing behavior prediction. Optimization of k-means code can be done using simulated annealing techniques. Simulated annealing is used to solve the problem of local minima. This function searches for global minimum of a very complex non-linear objective function with a very large number of optima. We are going to optimize k-means code by simulated annealing using the GENSA package present in the R-studio. After the clusters are made the system automatically recommends the clusters that are having the same browsing behavior patterns . Here we are analyzing the browsing habits of a group of users and these users are grouped into different clusters in the server so that we can predict these type of users who tend to browse the similar types of websites. The main idea used in the system is that it will recommend the websites that the users are frequently visiting. The implementation phase includes the entire system being loaded into a cloud framework of different users and a cloud server which is based on Google App Engine. The user belonging to the same cluster are recommended based on the browsing behavior. In this paper we focus on the optimization of k-means code using simulated annealing techniques. In order to get the accuracy, we just compare the browsing behavior of different users in the database and the websites the system will be going to recommend.. With our project we try to implement a movie recommendation system[15] which would provide the users with movie suggestions using Collaborative Filtering and Clustering algorithms building a model from a user's past behavior (movies previously watched or selected and/or numerical ratings given to those movies) as well as similar decisions made by other users.*

***Index Terms:**Clustering, GENSA, Google app engine, Recommendation system*

I. INTRODUCTION

Web data mining can be considered as the application of data mining used in the extraction of knowledge from the web data. There may be large number of web documents. Web usage mining mainly focuses on retrieving interesting patterns. Web content mining can be used in the extraction of useful information from the web documents. The browsing behavior of many users can be predicted using the recommendation systems[5]. These systems are supposed to be called as interactive softwares. Users preference to any items can be rated using Recommendation systems[3].So these systems are categorized to a class of information filtering system. The applications of recommendation systems includes movies, music and websites. These systems can also be used in real life applications.

Recommendation system mainly uses two different techniques. Such as collaborative filtering and content based filtering techniques for producing the list of recommendations. The browsing behavior of users can be predicted using the recommendation systems. This paper discusses the k-means code optimization using simulated annealing.

II. OBJECTIVE

This project shall focus on development and evaluation of a Recommendation System within the Movie Suggestion domain.

We propose a novel Collaborative Filtering based Recommendation System built specifically to produce high-quality Recommendations using tailored and improved clustering algorithms and tools. Recommender systems plays an important role in consumers lives. Recommender systems mainly help the users to buy the items that they prefer to buy based on the large amounts of data they have collected. The social networking mediums mainly uses these types of recommendation systems. Mainly recommendation systems concentrates on parsing large amounts of data to predict a users preference is the ultimate aim of the recommender systems. The hybrid filtering approaches such as collaborative filtering and content based filtering are the main approaches behind the recommendation systems.

The main aim is to apply a collaborative filtering algorithm in a rating website that has users and movies information, such as location, item's information, such as genre and release year, as well as ratings for items by users. Many algorithms are being applied on data to predict a user preference. There are many methods of predicting a user preference such as User-based, Item-based, and Model-based .The function performance will depend on the number of users, items, or clusters in each one respectively.

III. DATASETS AND PRE PROCESSING

The first and foremost task is to collect the data set which we will be operating upon. This dataset derived from Movielens describes 5-star rating and free-text tagging activity . It also contains 100004 ratings and 1296 tag

Revised Manuscript Received on June 10, 2019.

Sowmya K Menon, Research Scholar, Bharathiar University, Coimbatore, T.N, India.

Varghese Paul, Professor in IT, CUSAT,Cochin, Kerala, India

*Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication*

applications across 9125 movies.

Around 671 users created these data between January 09, 1995 and October 16, 2016.

The files contain different types of data such as
ratings.csv
tags.csv
links.csv, and
movies.csv,

The data set connects the user rating with the movie Id , each user also has a specific Id.

userId, movieId, rating, timestamp are included in the rating.csv

UserIds: The users named MovieLens users were randomly selected. Anonymized ids have been used. There should consistency between the user ids ratings.csv and tags.csv. (i.e., the same id refers to the same user across the two files).

A. Ratings.csv

The ratings of different movies are given below. Different movies have different ratings. The ratings are mentioned in the file ratings.csv. Each line of this file after the header row represents one rating of one movie by one user, and has the following format:

User Id, movie Id, rating, timestamp The ordering of lines within this file are made first by user Id, then, within user, by movie Id.

Ratings are made according to a 5-star scale, with half-star increments (0.5 stars 5.0 stars).

	userId	movieId	rating	timestamp
15268	99	4993	5.0	1044786919
15269	99	4995	5.0	1015037662
15270	99	5013	4.0	1015037662
15271	99	5349	4.0	1025915895
15272	99	5952	4.0	1044786919
15273	99	5991	5.0	1044754204
15274	100	1	4.0	854193977
15275	100	3	4.0	854194024
15276	100	6	3.0	854194023
15277	100	7	3.0	854194024
15278	100	25	4.0	854193977
15279	100	32	5.0	854193977
15280	100	52	3.0	854194056
15281	100	62	3.0	854193977
15282	100	86	3.0	854194208
15283	100	88	2.0	854194208
15284	100	95	3.0	854193977

B. Movies.csv

The title of the movies and the corresponding movie ids are being given. Movie Ids: Only
Movies

With AtleastOne rating or tag are included in the dataset. There should be consistency in the movie ids with those used on the Movie Lens website (e.g.,id1corresponds to the URL https://movielens.org/movies/1). Movie ids are found to be consistent between ratings.csv, tags.csv, movies.csv, and links.csv (i.e., the same id refers to the same movie across these four data files).

The Movie informations are contained in the file movies.csv. Each line of this file after the header row represents one movie, and has the following format:

movieId,title,genres Movie titles are entered manually or imported from https://www.themoviedb.org/, and include the year of release in parentheses

	movieId	title	genres
1	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
2	2	Jumanji (1995)	Adventure|Children|Fantasy
3	3	Grumpier Old Men (1995)	Comedy|Romance
4	4	Waiting to Exhale (1995)	Comedy|Drama|Romance
5	5	Father of the Bride Part II (1995)	Comedy
6	6	Heat (1995)	Action|Crime|Thriller
7	7	Sabrina (1995)	Comedy|Romance
8	8	Tom and Huck (1995)	Adventure|Children
9	9	Sudden Death (1995)	Action
10	10	GoldenEye (1995)	Action|Adventure|Thriller
11	11	American President, The (1995)	Comedy|Drama|Romance
12	12	Dracula: Dead and Loving It (1995)	Comedy|Horror
13	13	Balto (1995)	Adventure|Animation|Children
14	14	Nixon (1995)	Drama

c. Tags.csv

Tags Data File Structure (tags.csv): All tags are contained in the file tags.csv. Each line of this file after the header row represents one tag applied to one movie by one user, and has the following format:

userId,movieId,tag,timestamp The lines within this file are ordered first by userId, then, within user, by movieId.

Tags are used to make user-generated metadata about movies. Each tag can be represented as a single word or short phrase. The user determines the meaning, value, and purpose of a particular tag.

	userId	movieId	tag	timestamp
1	15	339	sandra 'boring' bullock	1138537770
2	15	1955	dentist	1193435061
3	15	7478	Cambodia	1170560997
4	15	32892	Russian	1170626366
5	15	34162	forgettable	1141391765
6	15	35957	short	1141391873
7	15	37729	dull story	1141391806
8	15	45950	powerpoint	1169616291
9	15	100365	activist	1425876220
10	15	100365	documentary	1425876220
11	15	100365	uganda	1425876220
12	23	150	Ron Howard	1148672905
13	68	2174	music	1249808064
14	68	2174	weird	1249808102
15	68	8623	Steve Martin	1249808497
16	73	107999	action	1430799184
17	73	107999	anime	1430799184

C. Links.csv

Links mainly contain the identifiers that can be used to link to other sources of movie data are contained in the file links.csv. Each line of this file after the header row represents one movie, and has
the following format:

movieId,imdbId,tmdbIdmovieIdisanidentifierformoviesusedbyhttps://movielens.org. E.g., the movie Toy Story has the link https://movielens.org/movies/1. imdbId is an identifier for movies used by http://www.imdb.com.

E.g., the movie Toy Story has the link
http://www.imdb.com/title/tt0114709/.
tmdbIdisanidentifierformoviesusedbyhttps://www.themoviedb.org. E.g., the movie Toy Story has the link https://www.themoviedb.org/movie/862.

The movies having the maximum ratings are being recommended to the active users. Some movies are having the same ratings and that type of movies are made a cluster and being recommended to the active users. Similar cases of recommendation systems can be used in the case of browsing behavior of websites. Some people are more prone to browse social networking sites .Some may be having the tendency of browsing the sports sites etc. Users having same type of browsing behaviours are recommended and they have being grouped as similar clusters.

Some may be the frequent browsers of news channels.

	movieId	imdbId	tmdbId
13	13	112453	21032
14	14	113987	10858
15	15	112760	1408
16	16	112641	524
17	17	114388	4584
18	18	113101	5
19	19	112281	9273
20	20	113845	11517
21	21	113161	8012
22	22	112722	1710
23	23	112401	9691
24	24	114168	12665
25	25	113627	451
26	26	114057	16420
27	27	114011	9263
28	28	114117	17015
29	29	112682	902
30	30	115012	37557
31	31	112792	9909

IV. MODULAR DESIGN OF THE PROJECT

The project is divided into two modules clustering and recommendations. Clustering is the process of grouping of similar data items. The client server architecture is the main architecture used in the system. Users belonging to the same browsing behaviours are clustered and are stored in the server. The clustering of similar users and the recommendation systems will take place in the server. Extensions helps to modify and helps in increasing the functionalities of the chrome browser. They are also considered to be small software programs. They can written using web technologies such as HTML, Java script and CSS.

Google App Engine is a platform for developing and hosting web applications in Google-managed data centers. One of the data storage facilities that the google uses is the Big table. Big table is considered to be a distributed, persistent, multidimensional sorted map. Results have proven that the Big table is not a relational database. Google App Engine facilitates running the web applications on Google's infrastructure App.

Engine applications are easy to build, easy to maintain, and easy to scale as your traffic and data storage needs grow. With App Engine, there are no servers to maintain: You just upload your application, and it's ready to serve your users.

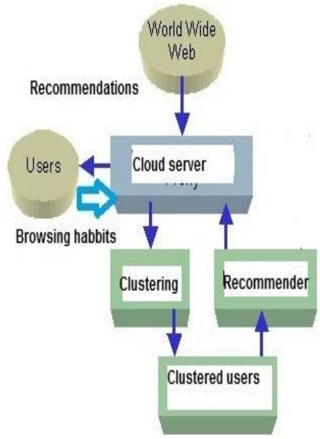


Fig -1: Architecture diagram

The above figure shows the architecture for the recommendation systems[7]. When the user browse the websites, it is the role of the client part to collect the information and submit it to the server periodically. The server functions as a cloud based server and here is the place where the actual clustering takes place. The websites[7] that are used by the clustered users are returned to the client plug-in by the recommendation systems. The user proceeds with further browsing based on these recommendations[17]. The web applications are run by the google app engine. The advantage of using app engines are it is easy to build and easy to maintain. There are no servers in the google app engine. By uploading your application it can be ready to be used by users.

In the implementation phase, this system was tested among, 100 users, by keeply observing their browsing habits for 1 month. Plug Ins are installed from the web site and their browsing habits may be observed for two weeks. It is to be proved that the users belonging to the same cluster as returned by the software, should be visiting the same web sites in next two weeks as suggested by the recommendation system.

V. IMPLEMENTATION OF K-MEANS USING THE IN BUILT PACKAGE ON THE STANDARD DATASETS

Cluster Analysis or **Clustering** is the grouping of set of objects such that objects in the same group (called a **cluster**) are more similar to each other than to those in other groups referred to as clusters.

K-Means Clustering Algorithm[4]

Input: we have to input '*k*',*as the number of clusters to be partitioned; 'n', as the*

number of objects.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Output: A set of '*k*' *clusters that is based on given similarity*
function.

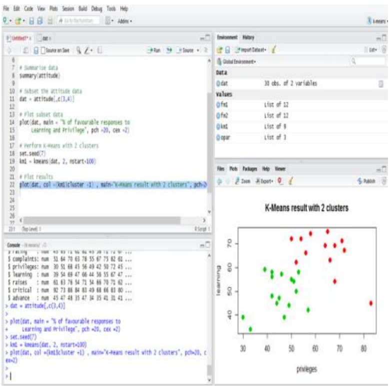
Steps:

(i) Arbitrarily **choose '*k*' *objects*** *as the initial cluster* centers.

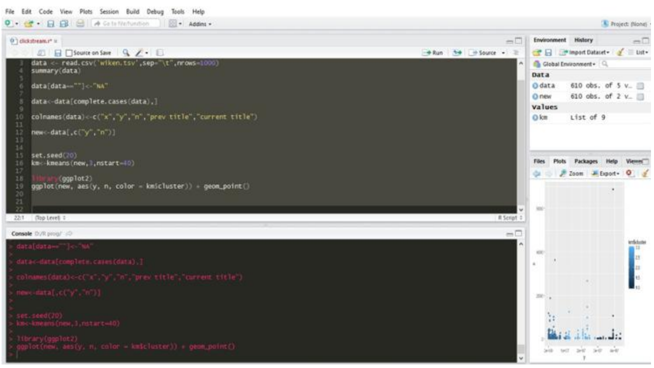
ii) Repeat,

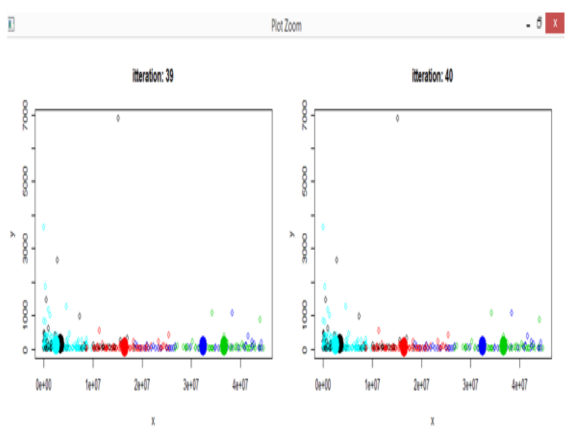
a. (Re) assign each object to the cluster to which the object is the most similar; based on the given similarity function;
b. We have to do the updation of the centroid, i.e., **calculating the mean value** of the objects for each Cluster.

iii) Repeat these steps until no change occurs in the mean value.



VI. K-MEANS CLUSTERING OF CLICK-STREAM DATA





k-means code was developed in R-Studio environment with a customizable distance function.

```
myEuclid<-        function(points1,        points2)
 {
  distanceMatrix<-  matrix(NA,  nrow=dim(points1)[1],
ncol=dim(points2)[1])
  for(i              in              1:nrow(points2))
        {
                  distanceMatrix[,i]                  <-
sqrt(rowSums(t(t(points1)-points2[i,])^2))
                  }
distanceMatrix
}
K_means<- function(x, centers, distFun, nItter)
 {
clusterHistory<- vector(nItter, mode="list")
centerHistory<<- vector(nItter, mode="list")
for(i in 1:nItter)
{
                  distsToCenters<- distFun(x, centers)
                  cluster<-     apply(distsToCenters,     1,
which.min)
                  centers<-  apply(x,  2,  tapply,  cluster,
mean)
                  # Saving history
                  clusterHistory[[i]] <- cluster
                  centerHistory[[i]] <- centers
                  }
  list(cluster=clusterHistory, centers=centerHistory)
  }
```

Next we are trying to optimize the K means code by using Simulated Annealing using the GenSA package present in the R studio.

Usage

GenSA(par, fn, lower, upper, control=list(), ...)

VII. SIMLUATED ANNEALING

SimluatedAnnealing[2] is used to solve the problem of local minima. This function is searching for global minimum of a very complex non-linear objective function with a very large number of optima.

Usage

GenSA(par, fn, lower, upper, control=list(), ...)

VIII. K-MEANS OPTIMIZATION

- Initial cluster number[5] and seed value has a very considerable effect on the clusters formed at the end of iteration. Different combination of the number of cluster and seed value will result into different sse value.
- SSE –Squared Sum Error , this is the value used to measure the efficiency of the cluster . It tells how well the clusters are packed together. It measures the distance of each cluster point with respect to its cluster centres using a distance function.
- Value of the cluster SSE is found corresponding to the different number of clusters and varying seed values.

IX. SIMULATED ANNEALING BASED

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

CLUSTERING PROCESS& RESULTS

K Means Clustering aims to partition observations into k clusters in which each observation belongs to the cluster with the nearest mean, serving as a prototype of the cluster. This results in a partitioning of the data space into Voronoi cells.

The problem is computationally found to be difficult (NP-hard) however, there are efficient heuristic algorithms that are commonly employed and converge quickly to a local optimum.

Let (x1, x2, ...,xn),be a set of observations such that each observations are considered to be a d-dimensional real vector.

The main aim of k means clustering is to partition the no of observations into k (n) sets S = S1, S2, ..., Sk so as to minimize the within-cluster sum of squares(WCSS)(sum of distance function.

Of each point in the cluster to the K center). In other words, its objective is to find: where i is the mean of points in Si.

Algorithm:

LetX=x1,x2,x3,........,xn be the set of datapointsandV=v1,v2,.......,vc

be the set of centers.

Step 1: Randomly select ‘c’ cluster centers

Step 2: Calculate the distance between each data point and cluster enters.

Step 3: Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers.

Step 4: Recalculate the new cluster center using the formula shown above:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} (\|x_i - v_j\|)^2 \quad (1)$$

where, ‘ci’ represents the number of data points in ith cluster.

Step 5: Recalculate the distance between each data point and new obtained cluster centers.

Step6: If no data point was reassigned then stop, otherwise repeat from step 3.

K-means algorithm is very fast, robust, scalable and easier to understand.It is considered to be very simple. Its time complexity is O(nkt), where N can be represented as number of objects, k is the number of clusters and t is supposed to be the number of iterations.There are many draw backs of this algorithm and this can be further modified in this future.

- The final results of K-means algorithm depends mainly on initial clustering centers. Different seed-points can form different clusters, and can even lead to no resolution.
- The algorithm concludes with the local minima value.

If the K clustering centre is named as cent pattern and pattern sample category vector named patterns are considered to be an ideal solution for the clustering problem, the distance is Euclidean distance the clustering problem is optimization problem, and the goal of the clustering problem is to minimize the sum of all distances between each pattern sample and the closest clustering centre, that is, the objective function is the minimization of sum of distances:

$$D = \sum_{i=1}^{K} \sum_{x \in C_i} dist(c_i - x)^2. \quad (2)$$

Here, Ci is the ith cluster, ci is the ith cluster centre, xi s a pattern sample in Ci, dist denotes the euclidean distance between two objects.

The details of the new clustering algorithm can be given by the following steps:

Step 1: Initializing the vectors for specific range of seed values and cluster values.

Step 2: For each seed value an evaluate function is run to calculate the ssei.e sum of square error .The sse value is calculated for different cluster values. The best combination of seed and the cluster for minimal sse value is chosen.

Step3:Inorder to provide optimization for the result of k means algorithm simulated annealing techniques can be used to minimize the net result of the function to evaluate. We have been using GenSA package in R for the implementation of simulated annealing process. GenSA , mainly searches for global minimum of a very complex nonlinear objective functions with a very large number of optima. GenSA(par, fn, lower, upper, control=list(), ...) is the usage format for GenSA. par - Vector. Initial values for the components to be optimized. Default is NULL, in which case, default values will be generated automatically

Fn –A function should be minimized, with first argument the vector of parameters over which minimization is to take place. The function should return a scalar result. lower - Vector with length of par. Lower bounds are used for components. upper - Vector with length of par. Upper bounds are used for components We have used the evaluation function as a parameter for GenSA , the seed value and the cluster value are the two values(par) to be optimized.

Step 4: Gradually each value is analysed and the best combination resulting from the above operation is displayed which can be used as the seed value and the cluster value for the recommendation engine to get the best recommendations[12]. As per the concept of simulated annealing the temperature decreases According to the cooling function, So here evaluate function act as the cooling function and sse value is minimized and the parameters determining the sse value are the cluster and seed value.

Step 5: The above process repeats itself for the entire range of the two parameters i.e the cluster and seed number till the upper limit is reached, for the maximum allowed time. After the completion of the entire process the best combination of the factors affecting the temperature is the resulting value.

After performing the above steps on the movie lens data set being used in the implementation a graph was generated using the gg plot package in R. The graph maps the squared sum error value on the y axis and the seed and cluster parameter on to the x axis .This graph helps to compare the result in the neighbouring cluster for different seed value. this minimum sse value for the parameter gives the most efficient result. So the above combination of seed and cluster value is selected for k means clustering algorithm that would give the most optimized result for recommendation system. The graph below shows the plot for the seed and cluster values.

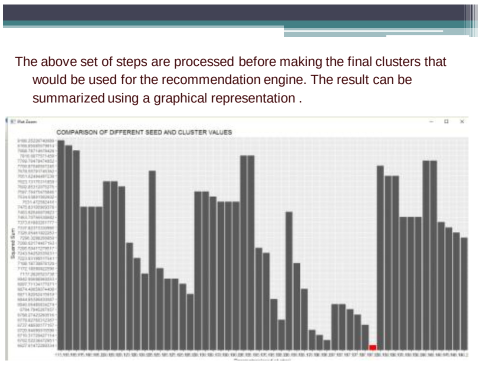


Fig -2: illustration of final clusters that would be used for the recommendation engine

X. RECOMMENDATION

Recommendation systems or Recommendation Engines are considered are subclass of information filtering systems that seek to check the "ratings" or "preferences" that a user would give to an item and recommends other products based on previous logs. Recommendation systems[6] have become extremely common in recent years, and are utilized in a variety of areas such as Movie Review sites, Music streaming sites, and Products on ecommerce websites[11].

Recommendation systems [13]uses a number of different technologies. We can classify these systems into two broad groups.

- Collaborative Filtering –The designing of recommendation systems can be facilitated using collaborative filtering. Collaborative filtering [6]helps in collecting and analyzing large number of informations based on users behaviour and helps in predicting what are the similarities That the user will be having to other users. The main advantage of collaborative filtering [8]is that it doesn't depend on machine analyzing contents and it has great capability in analyzing complex items like movies without getting an idea of it. In future many algorithms can be used to find similarirties.

For designing of recommendation systems another common approach used is content based filtering. This approach mainly deals with describing an item and a profile having user's preference. Content based filtering mainly uses keywords to describe items and user profile indicates the item s that the user likes. In content based filtering, many algorithms are developed for recommending items that are similar to those that a user liked in the past. This approach compares various items that are previously rated by the user and the best from these items are recommended. This approach also helps mainly in information retrieval[10].

For implementing the features of the items in the system, an item presentation algorithm can be used. A most widely used algorithm for the item presentation is the tf–idf representation (also called vector space representation). To create a user profile, the system mainly concentrates on two types of information:

1. To build a model that is of the user's preference.
2. Maintaining a history based on the user's interaction with the recommendation system.

Basically, these methods uses a set of attributes and special features for representing items withthin the system.. The system automatically creates a content-based profile of users depending upon the weighted vector of item features. The weights represents the relevance of each feature to the user and can be computed by individual rating content vectors.. Approaches such as decision trees, Bayesian classifier ,artificial neural networks etc can be used for estimating the probability that a user has like for a particular item

The recommendation processs[10] can be carried out by clustering the movie into number of clusters and the specific seed value can be obtained by the above process.

- auserID for which the Recommender[3] recommends movies
- and the number no_films of movies to recommend

Step1: Our first function(clusterFilms) clusters the movies based on their genre affiliation using k-means

Step2: getUserInfo<-function(dat,id) , returns all the movies with the associated ratings our selected user has already watched, The return value is an activeUser data frame with the columns "itemid"(=movieid), "rating" and "cluster".

Step3: setUserFilmCluster<-function(movieCluster, activeUser)

It assigns to each movie the corresponding cluster number.

Step4: getMeanClusterRating<-function(movieCluster, activeUser)

For each cluster the average of the movie ratings is calculated, the return value is "like", an integer vector, in which all clusters whose average rating is greater or equal 3 are included. If we do not find clusters with >= 3 rating, we give back a dummy value of zero.

Step5: getGoodFilms<-function(like, movieCluster, titleFilmDF)

Among the clusters which have a greater or equal "3" rating, we select the highest rating cluster , search movies (both active user has and has not yet watched) of this cluster. If there is no cluster with a baseline rating of 3 or above, random 100 movies are selected.

Step6: getRecommendedFilms<-function(titleFilmDF, userDF, userid)

Select all movies our user has not yet seen. The return value contains in addition to the movie ID the movie title by invoking the above functions.

Step7: suggestFilms<-function(titleFilmDF, userDF, userid, no_films)

This function recommends a particular user (userid) a certain number (no_films) of movies.

As an output we get the list of movies that is to be recommended to the Active user.

XI. CONCLUSION

In these recent years, recommendation systems have become very important. These systems have been used in many areas of our real life. Researches have proven that a hybrid approach including collaborative and content based approaches found to be more effective in recommendation systems[12].In order to implement the hybrid [14]approaches content based and collaborative filtering predictions are made separately and combined the content based to a collaborative approaches.

XII.ACKNOWLEDGMENT

I sincerely thank my guide Dr. Varghese Paul ,Professor in IT Cochin University of Science and Technology, who assisted me to do the research work in a successful manner.

XIII. FUTURE ENHANCEMENTS

Big data analysis for huge data sets by making use of large data sets. Now only dealing with small data sets.

REFERENCES

1. "**Dr. M. SudheepElayidom**" Data Mining and Warehousing" Cengage Learning 2015 ISBN-13 978-81-315-2586-9
2. "**G. PhanendraBabu**", **M. NarsimhaMurty**" Simulated annealing for selecting optimal initial seed in the k means algorithm", 199
3. "**GediminasAdomavicius and Alexander Tuzhilin**" Toward the Next Generation of Recommender Systems: ASurvey of the State-of-the-Art and Possible Extensions , ieee transactions on knowledge and data engineering, vol. 17, no. 6, june 2005
4. K-Means Algorithm Based on Real Data , Proceedings of the Federated
5. Jia Li and Osmar R. Ziane, "Combining Usage, Content, and Structure Data to Improve Web Site Recommendation***", In** *Proceedings of the 5th International Conference, EC-Web* , Spain, 2004, pp 305-315.Reference 1
6. Harita Mehta, ShvetaKundra Bhatia, PunamBedi and V. S. Dixit, (November 2011). Collaborative Personalized Web Recommender Systemusing Entropy based Similarity Measure. *IJCSI International Journal of Computer Science Issues*, 8(6), pp. 231-240.
7. G. Adomavicius and A. Tuzhilin,(2005).Towards the next generation of recommenender systems: a survey of the state-of-the-art and possible extensions. *IEEE Trans. on Data and Knowledge Engineering*, 17(6), pp. 734-749.
8. Ungar L.H. and Foster D.P, "Clustering Methods for Collaborative Filtering", In the Proceedings of the Workshop on Recommender Systems, AAAI Press, 1998.
9. Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John Riedl, ".Analysis of recommendation algorithms for e-commerce", In*Proc of ACM Conference on Electronic Commerce*, Minneapolis, MN, USA , 2000, pp. 158–167.
10. UpendraShardanand and Patti Maes, "Social information filtering: Algorithms for automating "word of mouth"", In *Proc of ACM CHI'95 Conference on Human Factors in Computing Systems*, Denver, Colorado, USA , 1995, pp. 210–217.
11. W. Wong and A. Fu. Incremental document clustering for web page classification, 2000. [Online]. Available:http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.21.2379&rep=rep1&type=pdf
12. Arbee L.P, Chen Yi-Hung Wu, Yong-Chuan Chen, "Enabling personalized recommendation on the web based on user interests and behaviours", In *proc. of 11th International Workshop on research Issues in Data Engineering*, Heidelberg, 2001, pp.17-24.
13. JanuzSobecki ,"Implementations of web based recommender systems using hybrid methods", In International journal of computer science and applications,Vol.3 issue 3,pp 52-64.
14. C. Lin, S. Alvarez, and C. Ruiz. Collaborative recommendation via adaptive association rule mining, 2000.
15. Xiaobin Fu, Jay Budzik, and Kristian J. Hammond. Mining navigationhistory for recommendation. In *Intelligent User Interfaces, pages 106–112,*2000.W.

AUTHORS PROFILE

Ms. Sowmya K Menonholds MCA from SNR sons college, Coimbatore under Bharathiar university. She is currently doing her research in computer science in the field of data mining at Bharathiaruniversity Coimbatore. She has completed her bachelors degree in chemistry from MG University Kottayam..She has more than 12years of teaching experience..She has published articles in reputed national and international journals and has attended international conference also.

Dr. Varghese Paul obtained B.Sc (Engg) degree in Electrical Engineering from Kerala University, M.Tech in Electronics and Ph.D in Computer Science from Cochin University of Science and Technology.

His research areas are *Data security using Cryptography, Data Compression, Data Mining, Image Processing and E_Governance.* He is the developer of TDMRC Coding System for character representation and encryption system using this coding system. He has got many research publications in international as well as national journals. He has published a text book also.

He had been Professor and Head of Information Technology Department in Cochin University of Science and Technology and currently acts as Research Coordinator there. Also he isResearch Supervisor of *Kerala Technological University, MG University Kottayam, Anna TechnicalUniversity Chennai, Bharathiar University Coimbatore and Bharathidasan University Trichy.* 34 research scholars have already completed research studies under his guidance.

Earlier he has worked as Industrial Engineer with O/E/N India Ltd Cochin, Communication Engineer with KSE Board, SCADA Engineer in Saudi Electricity Department and Dean (CS, IT and Research) in Toc H Institute of Science and Technology.

He is a Certified Software Test Manager, Ministry of Information Technology, Government of India. Also, member of Information System Audit and Control Association, USA and Indian Society for Technical Education, India.