International Journal of Scientific Research & Engineering Trends
Volume 8, Issue 2, Mar-Apr-2022, ISSN (Online): 2395-566X

Friend Recommendation on Social Media

M.Tech. Scholar Atul Kumar Churhe, Asst. Prof. Girish Gogate
Department of Computer Science and Engineering
Rishiraj Institute of Technology (RIT),
Indore, India
atulchurhe@gmail.com, girishgogatee@mail.com

Abstract- Friend recommendation is one of the most popular characteristics of social network platforms, which recommends similar or familiar people to users. The concept of friend recommendation originates from socialnetworks such as Twitter and Facebook, whichuses friends-of-friends method to recommend people.We can say users do not make friends from random people but end up making friends with their friends' friends.The existing methods have narrow scope of recommendation and are less efficient. We put forward a new friend recommendation model to overpower the defects of existing system.For better friend recommendation system with high accuracy, we will use collaborative filtering method to compare similar, dissimilar data of users and will make a recommendation system which gives user to user recommendation based on their similar choices, activities and preferences. Location based friend recommendation system are becoming popular because it brings physical world to digital platform and gives better insight of user's preferences or interest This recommendation system will increase the scope of recommendation from one user to other with similar set of interest and their location.

Keywords- Friend recommendation, collaborative filtering, social network, Recommendation system.

I. INTRODUCTION

Friend recommendation is one of the most common and fundamental service in LSBN platform which recommends familiar or interested user to each other. About 71% of internet users were online social network users and they will grow in near future. Social networking is very popular online activities with high rate of user interactions & expanding mobile possibilities.

The growth rate in use of smart phones and mobile devices is very rapid and has opened up new areas of mobile social networks with increased features.With over billions of monthly active users on social network. Facebook is currently the market leader in terms of user engagement reach and scope [1].

Recent advances in localization techniques have improved social networking services, allowing users to share their locations and location-related contents. Such type of social networks is referred as location-based social networks (LBSNs). LBSNs are equipped with type of friend recommendation which utilizes user's historical location information.

Traditional friend recommender engines provide a user with promising candidates to make friends based on their profiles, social structure & interactions. Location information can improve the effectiveness of recommendations. The basic idea is that user location histories reveal choices, and thus users with similar location histories have similar choices & have higher probability to be friends [2].

Friend recommendation service is used for conventional social networks. But there are very less algorithms that exploit LBSN data in recommendation. Earlier methods generally use GPS information to find the resemblance between users. When compared to GPS information, check_in information gives more context depended information. Furthermore, most of the LBSNs collect check_in information than the GPS trajectory data. The objective of our proposed recommendation systems is to include user profiles, interest, and user location histories (check_in data) and apply collaborative filtering methods for user to user recommendation to increase scope of recommendation and make it more efficient [3].

A location-based social network doesn't mean concatenating a location to an existing social network to allow people to share location related information and activities, but LBSN is also made up of the new social structure of individuals connected together by the inter dependency of their locations in the real world & their location-tagged media like text, image and video [4]. Physical location does not only include the instant point location of an individual at a given timestamp but the location history of an individual over a specified time period. Also, the knowledge, common interests, and preferred activities are derived from an individual's location information and location related content affects the social relations in LBSN [5].

LBSN is consists of a G <U, C> and social network G <U, E >. In G <U, E> U is the set of users and E is the set of edges which connects or indicates a social connection between different users in LBSN. In G <U, C> Check_in 'c' belongs to set C and shows user 'u' belongs to set U has a check in activity at location l at time t [2].

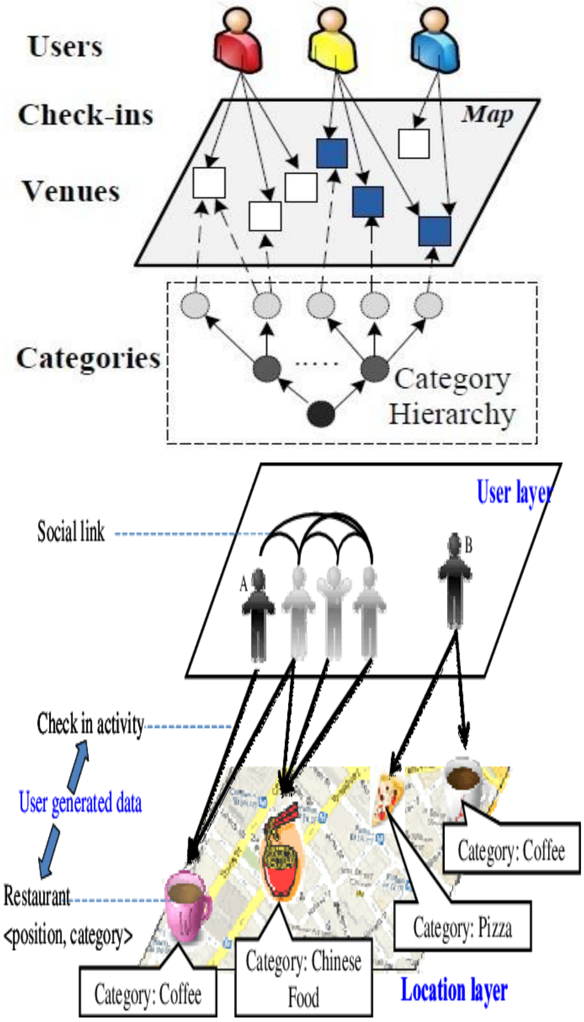


Fig 1. Location Based Social Network.

II. RELATED WORK

Temporal, spatial and social correlation is three main attributes of any LBSN. However, the situation which includes these three features cannot be solved in previous algorithms. There is no method which utilizes all information properly A new approach of friend recommendation is proposed, which aims to recommend friends with similar location preference for LBSN's users.

This approach first, use the method of local random walk based on Markov chain to calculate the user's friendship similarity on social network. Second, it calculates the user's location preference similarity in the real world based on check-in data and finally recommends friends to users by building a mixed user preferences model [6].

A new friend recommendation model (FE-ELM), is proposed where friend recommendation is regarded as a binary classification problem. In this model first feature extraction is done by using different strategies and then in training process ELM is selected as classifier to learn the the spatial-temporal feature, social feature, and textual feature, finally experiments are performed on real datasets for better efficiency and accuracy [7].

The new properties and challenges that location brings to recommender systems for LBSNs are discussed in this paper. First, author has categorized the recommender systemsby the objective of the recommendation, which include locations, users, activities, or social media. Second, they categorize the by the methodologies employed, including content-based, link analysis-based, and collaborative filtering. Then finally, classify the systems by the data sources used, including userprofiles, user online histories, and user location histories. For each category, the goals and contributions of each system are summarised and highlights the representative research effort. It introduces the concepts, unique properties, challenges, evaluation methods and future work for recommender systems in LBSNs [8].

Hierarchical-graph-based similarity measurement (HGSM) framework is proposed here, which models people's location histories and determines the similarity between users. In this framework, 3 factors sequence property of users' movements, Hierarchy property of geographic spaces, Popularity of different locations are considered. Using HGSM to estimate the similarity between users, a collaborative filtering-based method is also employed in our system to find an individual's interest in unvisited geospatial regions [9].

A friend recommendation algorithm is proposed which is known as Random walk-based context-aware friend recommendation algorithm (RWCFR). This model uses an undirected un-weighted graph that represents users, locations, and their relationships. RWCFR constructs a sub-graph according to the user's present environment.

Popular users and famous places in region are added to this sub- graph. After constructing the sub-graph, this sub-graph is given as input to algorithm, and it calculates the recommendation possibilities of users for suggesting becoming potential friend. A list of potential friends is generated according to output of the random walk algorithm [10].

Recommendation system make use of user profile, friend description and past behaviour for recommendation but no attention has been given to personalization based explicitly on social networks. Author has used information such as social graph among users, tracks & tags from last.fm social network which effectively incorporates bonds of friendship. We have done number of experiments betweenthe Random Walk with Restarts model and user-basedcollaborative filtering model. The results prove that the graph modelgains from the additional information implanted in socialknowledge [11].

The paper analyzes the main challenges of the collaborative filtering algorithm and provides several solutions. To solve cold start problem for the new user, we could replenish user's profile indifferent ways, the general approach is to require user providetheir profile while login the social account and for the new friend, we could combine the collaborativefiltering and content-based recommender algorithm.There are few solutions for the sparsity problem.The first one uses filling or decreasing the dimension to decrease the sparsity of the matrix. Another solution improves the efficiency of the algorithms without changing the sparsity of the matrix. [12].

III. PROBLEM DOMAIN

The traditional collaborative filtering recommendation algorithm is having lack of accuracy and efficiency as this uses formal method of filtering which makes it inefficient to use at alone. In terms of recommendation made by the collaborative filtering algorithm it may be concluded that the algorithm needs many more improvements.

By implementing traditional collaborative filtering recommendation algorithm, we get less accuracy which makes it typical to use and inefficient to apply on huge datasets i.e. Big Data. Dealing with big data the less accuracy makes it inappropriate and less accurate. As applying this algorithm on huge amount of data in real world applications the less accuracy will not be efficient for making recommendations to users.

The numbers of attributes which are available are totally considered for extracting information to recommend friends to users which makes the collaborative filtering recommendation algorithm inefficient. Also, the higher the number of attributes used to make recommendations, results in higher computing time and higher number of comparisons to be made. The overall dimensions included for making recommendation should be removed as per the requirement.

Apart from this the k-means clustering applied previously with the collaborative filtering algorithm can be replaced by different clustering technique. There are some drawbacks that can be seen in the k-means clustering technique which may be overcome by replacing this clustering technique with the newer one. In the k-means clustering the numbers of the clusters that should be made need to be defined at the start of the algorithm which makes it inefficient to use if the numbers of the clusters are not properly defined.

One more thing to be noted, that is the dimensionality of the given dataset should be less in number to lower the comparisons that will be made at the time of execution. The more the number of the dimensions to evaluate the results, makes the accuracy lesser and requires more time to make recommendations to the user. Hence to reduce the number of attribute or the dimensionality of the dataset is major task.

IV. PROPOSED METHOD

The problem observed in the previous algorithm can be removed by replacing the existing techniques by newer techniques. As in the previous, the algorithm combines the K-means clustering technique with the PCA as dimensionality reduction technique. Combining both this techniques in the collaborative filtering algorithm was a solution proposed earlier by the authors.

Here we have proposed a better clustering technique as compared to the k-means clustering, while keeping the PCA as earlier it was used. The k-means clustering can be replaced by the hierarchical clustering as it is better clustering technique to work on. The PCA will be used as the dimensionality reduction technique to decrease the dimensionality of the data.

The Hierarchical clustering will provide better results in comparison to the k-means clustering, as stated that in hierarchical clustering there is no need to define the number of clusters at the beginning of the clustering. Defining the required number of clusters after applying the hierarchical clustering will make it feasible to break the clusters as per the dataset. But before applying the clustering technique on the dataset the dataset should be improved. If the Input to the algorithm will be accurate then the obtained output will be more efficient. So, to improve the input dataset the dimensionality reduction should be done and to do this the PCA have to be applied on the dataset.

In final words we are going to apply the PCA on the dataset before giving it as input and after getting the principal components this are given as input to the hierarchical clustering. The collaborative filtering algorithm will firstly perform the PCA and after that the hierarchical clustering is applied and the final recommendations are made. Hence in this way the collaborative filtering algorithm can be improved and the recommendations can be made accurate.

1. Algorithm For Proposed Approach:

The proposed algorithm using both the techniques, the first one is the PCA which will help in reducing the dimensions of the given dataset and the second one is the clustering technique which is the hierarchical clustering. Here in our algorithm we are applying the PCA at first because it will reduce the dimensions of data and after that the hierarchical clustering will be performed on the obtained principal components. The working algorithm is as follows:

- **Step 1:** Data collection - collect the friend related data like name, rating etc. in the form of csv file.
- **Step 2:** Data pre-processing - perform manual data analysis and eliminate the feature which is less correlate to another feature.
- **Step 3:** Perform PCA (principal component analysis) on the data and save the data in to csv file.
- **Step 4:** Define hierarchical clustering (agglomerative) model.
- **Step 5:** Train the hierarchical clustering (agglomerative) model on the data.
- **Step 6:** Take the one user input and apply PCA on that.
- **Step 7:** Perform the prediction in the input it gives the cluster id.
- **Step 8**: Fetch all the friend detail which belong to this cluster id and make the list of it. (This list is recommended friend list)

2. Flowchart of the Proposed Approach:
Below we have given the flowchart for the proposed approach which will help in understanding the flow of the steps performed:

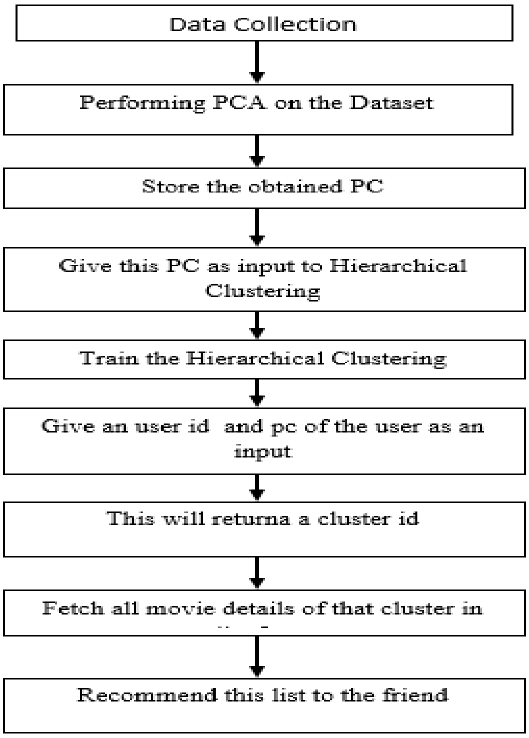

Data Collection
Performing PCA on the Dataset
Store the obtained PC
Give this PC as input to Hierarchical Clustering
Train the Hierarchical Clustering
Give an user id and pc of the user as an input
This will returna a cluster id
Fetch all movie details of that cluster in
Recommend this list to the friend

Fig 2. Flowchart for the proposed system.

3. Pearson Correlation Coefficient:
The Pearson correlation coefficient is a formula that measures the strength between variables and relationships. It is very helpful statistical formula is often referred to as the 'Pearson R' test. Whenever we want to find how strong relationship is between two variables, it is a good idea to apply a Pearson correlation coefficient test.

4. Formula:

In order to see how strong, the relationship is between 2 variables, a formula must be followed to produce what is referred to as the coefficient value. The coefficient value varies between -1.00 and 1.00. If the coefficient value is –ve, then it means the relationship between the variables is negatively correlated, and if the value is + ve, then it shows variables are positively correlated, or both values varies together either increase or decrease.

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2}}$$

Pearson Correlation Coefficient Formula

Note: The above examples only use data for 3 people, but the ideal sample size to calculate a Pearson correlation coefficient should be more than 10 people.

A comparision between different similarity calculation techniques is also discussed here which suggest why we have chosen Pearsons correlation.

Suppose we have 2 vectors x & y and we want to measure the similarity or degree of closeness between them. A basic similarity function is the inner product

Inner (x, y) = ∑ixiyi = ⟨x, y⟩

If x tends to be high where y is also high, and low where y is low, Higher the inner product vectors are more similar. The inner product is unbounded. A way to make it bounded between -1 and 1 is to divide by the vector's L2 norms which results in giving the cosine similarity .

CosSim (x, y) =∑ixiyi/√∑ix2i√∑iy2i = ⟨x, y⟩/ ||x|| ||y||

This is bounded between 0 and 1 if x and y are non-negative. Cosine similarity is not invariant to shifts/change. If x was shifted to x+1, the cosine similarity would change. Pearson correlation is invariant. Let Xand Y be the respective means:

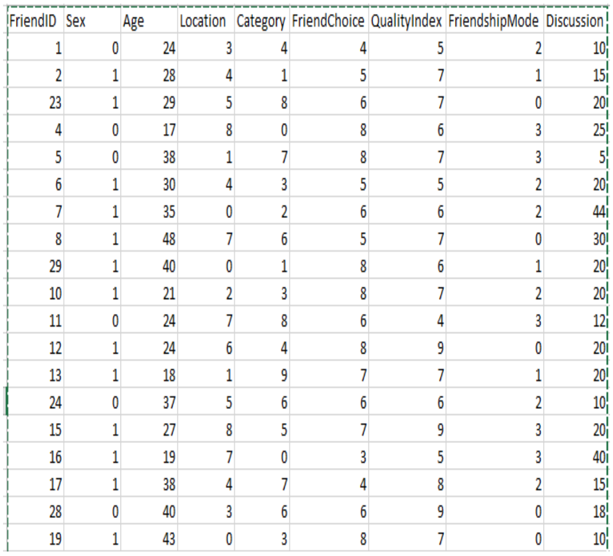
Corr (x, y) =∑i (xi− X) (yi− Y)/√∑ (xi− X)2 √∑(yi−Y)2

Correlation is the cosine similarity between centered value of x and y i.e mean value; it is also bounded between -1 and 1. People generally think about cosine similarity in terms of vector angles, but it can be not be used as a correlation, if you think of the vectors as paired samples then correlation is invariant to both scale & location changes of x and y.

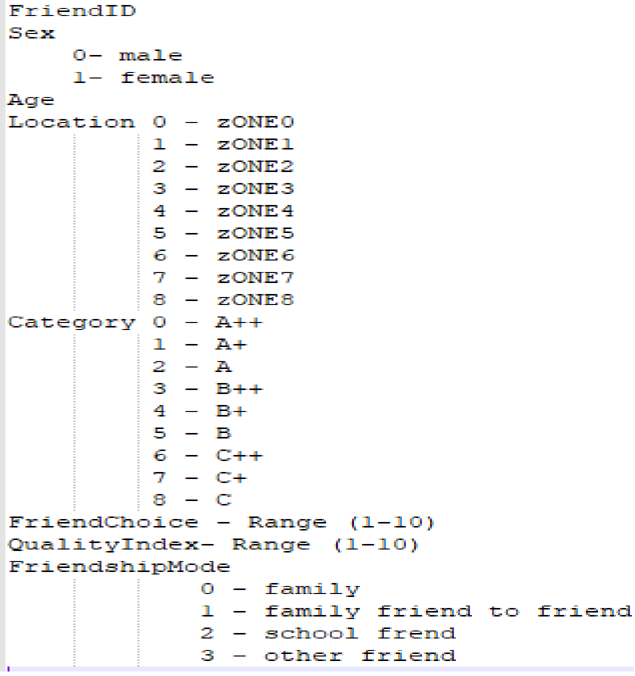
V. EXPERIMENTAL RESULTS AND EVALUATION

1. Data Set Processing and Experimental Result:
In this section, we implemented set of experiments that show for evaluating the impact of proposed system on recommendation. We have done different experiments on the Friend data set. In currently, we have a tendency to perform experiments on move choice knowledge collected from the friend recommendation web-based recommender system. The information set contained 600,000 choices from 824 users and one, 50 friends, with every user choice a minimum of twenty things on more details table 1 and figure 3**.**

Table 1. Data Set Attributes.



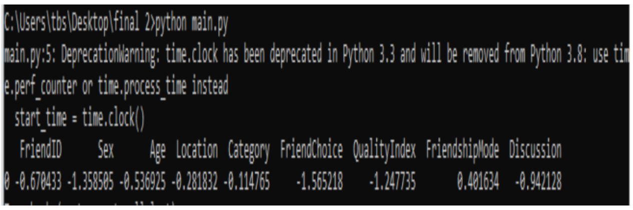
FriendID	Sex	Age	Location	Category	FriendChoice	QualityIndex	FriendshipMode	Discussion
1	0	24	3	4	4	5	2	10
2	1	28	4	1	5	7	1	15
23	1	29	5	8	6	7	0	20
4	0	17	8	0	8	6	3	25
5	0	38	1	7	8	7	3	5
6	1	30	4	3	5	5	2	20
7	1	35	0	2	6	6	2	44
8	1	48	7	6	5	7	0	30
29	1	40	0	1	8	6	1	20
10	1	21	2	3	8	7	2	20
11	0	24	7	8	6	4	3	12
12	1	24	6	4	8	9	0	20
13	1	18	1	9	7	7	1	20
24	0	37	5	6	6	6	2	10
15	1	27	8	5	7	9	3	20
16	1	19	7	0	3	5	3	40
17	1	38	4	7	4	8	2	15
28	0	40	3	6	6	9	0	18
19	1	43	0	3	8	7	0	10



```
FriendID
Sex
        0- male
        1- female
Age
Location 0 - zONE0
         1 - zONE1
         2 - zONE2
         3 - zONE3
         4 - zONE4
         5 - zONE5
         6 - zONE6
         7 - zONE7
         8 - zONE8
Category 0 - A++
         1 - A+
         2 - A
         3 - B++
         4 - B+
         5 - B
         6 - C++
         7 - C+
         8 - C
FriendChoice - Range (1-10)
QualityIndex- Range (1-10)
FriendshipMode
               0 - family
               1 - family friend to friend
               2 - school frend
               3 - other friend
```

Fig 3. Cleaning of Friend Dataset.

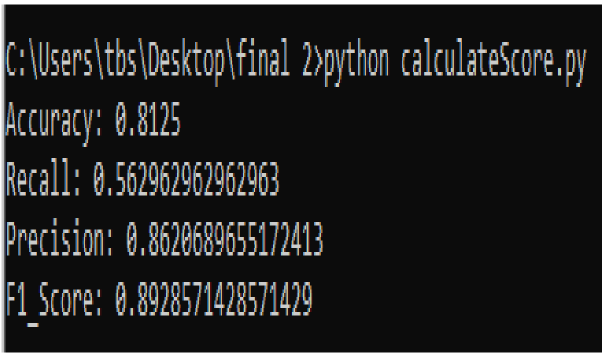
In this figure 3 cleaning of friend dataset. During cleaning we have clean all attributes like sex (0 to male and 1 to female); locations are dividing into zone wise 0 to 8, friend category is dividing into 0 to 8 and etc**.**



```
C:\Users\tbs\Desktop\final 2>python main.py
main.py:5: DeprecationWarning: time.clock has been deprecated in Python 3.3 and will be removed from Python 3.8: use tim
e.perf_counter or time.process_time instead
  start_time = time.clock()
   FriendID      Sex      Age  Location  Category  FriendChoice  QualityIndex  FriendshipMode  Discussion
0 -0.670433 -1.358505 -0.536925 -0.281832 -0.114765     -1.565218     -1.247735        0.401634   -0.942128
```

Fig 4. Display Data Set Attribute and it's Calculate Exaction Time.

In this figure 4 display all attribute on given data set like FriendID, sex, age, location, category, friend choice, quality index, payment mode and discount. Total exaction time taken 7.48 seconds.



```
C:\Users\tbs\Desktop\final 2>python calculateScore.py
Accuracy: 0.8125
Recall: 0.562962962962963
Precision: 0.8620689655172413
F1_Score: 0.8928571428571429
```

Fig 5. Display Accuracy, Recall, Precision and F1-Score on given Data Set.

In this figure 5 displays performance on given data set. Accuracy, recall, precision and f1-score are 81.25 %, 90.90%, 86.95 % and 86.95%.

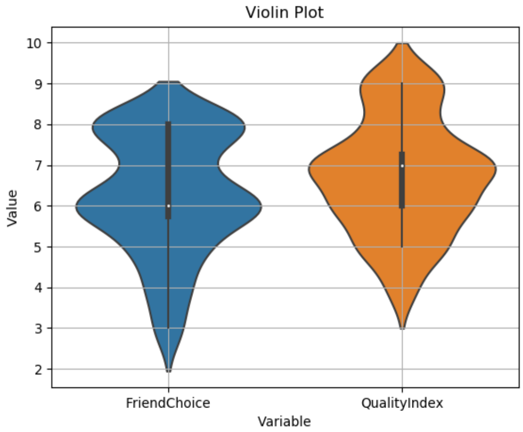


Fig 6. Violin plot between Friend Choice and Quality Index.

In this figure 7 displays density plot in all attributers on given data set attributes like age, location, category, friend choice, quality index, payment mode and discount.

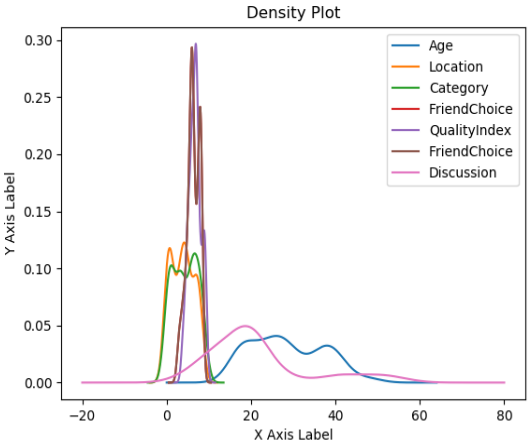


Fig 7. Density Plot in all Attributers on Given Data Set.

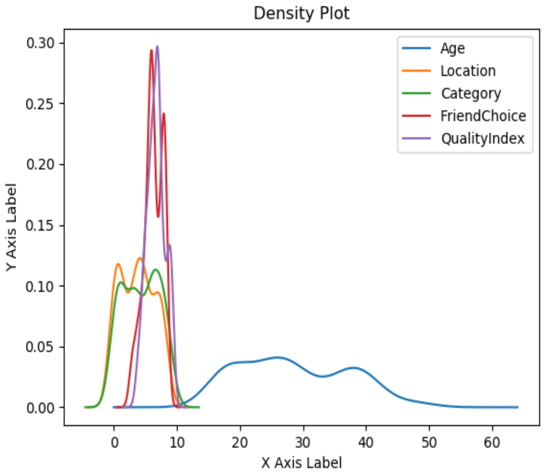


Fig 8. Density Plot in Some Attributers on Given Data Set

In this figure 8 displays Density Plot in Some Attributers on Given Data Set attributes like age, location, category, friend choice and quality index.

Developing a solution is an approach proving mechanism but to prove its results is a complicated task because it measures each and every step of the solution and let it compare with the existing mechanisms. So as to do that effectively this chapter gives a detailed result analysis to prove effectiveness of the suggested mechanism.

For making the analysis of the proposed approach we have used the Kaggle dataset the data about friends is taken from the Kaggle dataset and the friend_likes pattern and user details are combined from the Kaggle dataset. The experiment was carried out to evaluate the accuracy of the recommendations produced by the algorithm we have proposed in our paper. The accuracy term is calculated in this experiment by which the comparison between the proposed and the existing algorithm can be made.

We are applying this data on the previous collaborative algorithm with pca and k-means and the results are obtained, so the accuracy of the previous algorithm is calculated.

Accuracy = ({Relevant Document} intersection {Retrieved Document} / {Relevant Document}) *100

Now the proposed algorithm with hierarchical clustering is taken for analysis. The collaborative filtering algorithm along with pca and hierarchical clustering is analyzed over the same data. This algorithm's accuracy is compared with the existing algorithm.

The experiment clearly results in an increase in the accuracy of the recommendations made by our proposed algorithm. The results are compared between both the algorithms using k-means clustering with pca and hierarchical clustering with pca in terms of accuracy are shown in the following graph:

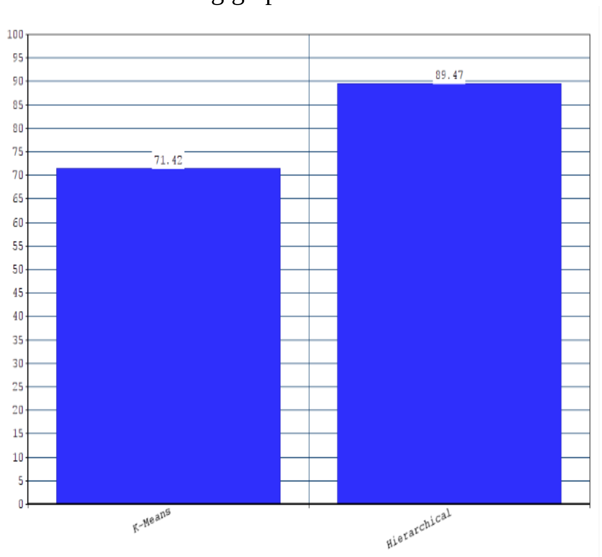


Fig 9. Accuracy results for both the algorithms.

The Fig.9 clearly concludes that the proposed hierarchical clustering works much better as compared to the previously used k-means clustering. The results in terms of accuracy of the proposed algorithm is higher than the earlier clustering technique. So it is better to use the Hierarchical clustering with pca on the collaborative filtering algorithm as compared to earlier one.

VI. CONCLUSION AND FUTURE WORK

The proposed research work observes the recommendations made by the system to the user. The entire work is done by the hierarchical clustering technique along with the pca, by which the accuracy of the system is evaluated.

The accuracy of the system is evaluated by the intersection of the recommended friends with the friend_likes made by the user for the friends earlier. The experiment shows better results from the earlier algorithms.

In future we can use other datasets to carry out the experiment. The other parameters apart from the accuracy can be tested. Different clustering technique may be applied to improve the algorithm.

REFERENCE

[1] Haruna K, Ismail MA, Damiasih D, Sutopo J, Herawan T. A collaborative approach for research paper recommender system. Plos One. 2017, 12(10):e0184516. https://doi.org/10.1371/journal.pone. 0184516 PMID: 28981512

[2] Rojas G, Garrido I. Toward a rapid development of social network-based recommender systems. IEEE Latin America Transactions. 2017, 15(4):753–759.

[3] Huang S, Zhang J, Wang L, Hua XS. Social Friend Recommendation Based on Multiple Network Correlation. IEEE Transactions on Multimedia. 2016, 18(2):287–299.

[4] Corbellini A, Mateos C, Godoy D, Zunino A, Schiaffino S. An architecture and platform for developing distributed recommendation algorithms on large-scale social networks. Journal of Information Science. 2015, 41(5):686–704.

[5] Fields B, Jacobson K, Rhodes C, Inverno M, Sanler M, Casey M. Analysis and Exploitation of Musician Social Networks for Recommendation and Discovery. IEEE Transactions on Multimedia. 2011, 13 (4):674–686

[6] Chamoso P, Rivas A, Rodrı ´guez Sara, Bajo J. Relationship recommender system in a business and employment-oriented social network. Information Sciences. 2018, s 433–434:204–220.

[7] Guo G, Zhang J, Zhu F, Wang X. Factored similarity models with social trust for top-N friend recommendation. Knowledge-Based Systems. 2017, 122:17–25.

[8] Zhang Z, Liu H. Social recommendation model combining trust propagation and sequential behaviors. Applied Intelligence. 2015, 43(3):695–706.

[9] Maier C, Laumer S, Eckhardt A, Weitzel T. Giving too much social support: social overload on social networking sites. European Journal of Information Systems. 2015, 24(5):447–464.

[10] Lee S, Koubek RJ. The effects of usability and web design attributes on user preference for e-commerce web sites. Computers in Industry. 2010, 61(4):329–341.

[11] AndreasenT, Jensen PA, Nilsson JF, Paggio P, Pedersen BS, Thomsen HE. Content-based text querying with ontological descriptors. Data & Knowledge Engineering. 2004, 48(2):199–219

[12] Huang S, Zhang J, Wang L, Hua XS. Social Friend Recommendation Based on Multiple Network Correlation. IEEE Transactions on Multimedia. 2016, 18(2):287–299.