International Journal of Allied Practice, Research and Review
Website: www.ijaprr.com (ISSN 2350-1294)

Interest Based Mobile Multimedia Recommendation System with the Help of Cloud-Based Technology

Dr. Prof. P. D. Lambhate and Ashish Das

**Department of Computer Engineering, JSPM's College of Engineering, Hadapsar, Pune, India
Savitribai Phule Pune University, Maharashtra, India.**

Abstract—in today's world many of the users are accessing internet among which 3/4th of them are the users accessing internet through mobile. As per emerging technologies, mobile bandwidth are having a lot of restrictions it may be due to network type, file size, bandwidth etc. Which is nothing but waste of time for mobile users to get their desired interests like photo bucket, orkut, movies villa etc. The proposed solution consists of a set of algorithms that identify and utilize unique item-based interest clusters and cluster-based item rating in order to recommend newly-generated content items to individual users in real time. Our main contributions are (1) a detailed analysis of content popularity (2) an interest-based clustering and cluster-based content recommendation solution and (3) a complete implementation and deployment in mobile application with help of cloud technology. Demonstrated the proposed system outperforms three widely-used collaborative filtering algorithms (SCA, graph partition, and k-means separately) in existing recommender systems. It can effectively identify personal interests and improve the quality and efficiency of real-time personalized content recommendation in communities. Hence, when a new user requests for any video then rules will match to make real-time recommendations. Where results will recommend the services with which accurate precision, accurate recall and a low response delay.

Keywords:-Cloud computation, minimal spanning tree, multi-media service recommendation, user behavior analysis, Recommender Systems, Social Networks, Friend recommendations, Community discoveries, Activity recommendations, Social media recommendations

I. INTRODUCTION

In this paper we tried to propose a mobile multimedia recommendations system which is based on user behaviour. Implemented on Ecliptics cloud where we are having a huge requirement which suggest or recommends as per the user behaviour. Below are the differences which is their with traditional systems are :

1) The collector and user behaviour are getting decentralized into different computing nodes.

2) In this case captured user behaviour clusters rather collecting only user profiles.

3) Approaching the graph-based optimizing mechanism which speeds up the recommendation process.

Following are the contributions for the proposed system:

1) The collector and user behaviour are getting collected rather collection of detailed user profiles. Here as much users are viewing will give and exchange with increment of collectors. But due maximum users visit might we face the network overhead issue, for its avoidance clustering of user-behaviour is used where collector calculates user clusters as per clustering rules which reports the user cluster to recommender only.

2) Since, today's multimedia recommendation system approached the cloud technology. In which, user clusters and different multimedia content are collected, distributed and stored accordingly. As discussed in user content recommendation, data are getting partitioned into several small chunks, where chunks are processed then; results are getting reduced and merged. Where the concept of MapReduce occurs which speeds up the different existing algorithms.

3) To improve the scalability and the real-time recommendation, the recommendation rules are sorted or ordered. Where the existing recommendation systems [11] always follow the ranked list by arranging the dummy data. Whereas approached work, as per recommendation rules the recommender searches a real-time ranked list of users. Also approaching a graph based rule applied to reduce searching latency.

II. LITERATURE SURVEY

In current years many tried to conclude the functionality and interests of different human behaviors, taking help of cloud technology where we can modify or construct a large amount of data and maintain it easily.

Poonam [11], Multidimensional data on semantic web proposed the tremendously with more and more information on the web being available in the form of Resource Descriptor Framework (RDF). Addition to that also was proposed [10] the feature extraction technique using semantic based crawler for search engine which resulted in convergence powerful analytical technologies, namely the Semantic web. JieBao[4] did a survey of recommender systems by analyzing

1) The data source used, 2) the methodology employed to generate a recommendation and 3) the objective of the recommendation. Where he proposed three taxonomies that partition the recommender systems according to the properties listed above. First, categorized the recommender systems by the objective of the recommendation, which can include locations, users, activities, or social media. Second, categorized the recommender systems by the methodologies employed, including content-based, link analysis-based, and collaborative filtering-based methodologies. Third, categorized the systems by the data sources used, including user profiles, user online histories, and user location histories. For each category, he summarized the goals and contributions of each system and highlighted one representative research effort. Further, provided comparative analysis of the recommendation systems within each category. Collaborative Filtering (CF) algorithms are widely used in a lot of recommender systems; however, the computational complexity of CF is high thus hindering their use in large scale systems. Zhi-Dan Zhao [7] implemented user-based CF algorithm on a cloud computing platform, namely Hadoop, to solve the scalability problem of CF. Where it shows that a simple method that partition users into groups according to two basic principles, i.e., tidy arrangement of mapper number to overcome the initiation of mapper and partition task equally such that all processors finish task at the same time, can achieve linear speedup.

III. PROPOSED APPROACH AND DESIGN

A. Problem Definition

In today's data traffic environment, need to proposed approach which can recommend desired services with high precision, high recall and low response delay.

B. Proposed Architecture

As mentioned earlier due to maximum mobile users in today's environment. Recommendation system are based on user's preferences which are based on user favorite recommendation which is been applied. We'll go through the system which is proposed in present and architecture of proposed cloud recommendation system. Let's check the following points in details:

1) **Preferred System:** If we go through the recommendation system where it focuses only on some specific domains. For example, Google News will suggest personalized news. Similarly, Amazon recommender system will suggest their desired products. And YouTube uses user watching history to suggest the videos for users. Similarly, many examples are same. Hence, in a general terms we can categorize it into four different categories of algorithms like: CB recommendation, CF-based recommendation[3], context-aware recommendation[7], and graph-based recommendation. Lets discuss in detail one by one :

CB Recommendation: System makes recommendation on following like content titles, tags, or descriptions. Where some searches or finds the user-interested items based on individuals recent history specifically as content. Its easy to implement but in some cases the user's profile information by a bag of words are not enough to collect the accurate interests of the user.

CF-Based Recommendation: System makes recommendation on user transaction histories and the popularity of content. For example, individual user's interests are predicted by a group of similar users. Now to obtain a content rating the methods like statistics or feedback are used. Hence, its been told that CF system requires enough historical consumption record and feedback.

Context-aware recommendation: System provides a stable recommendation without considering user context information. Its a dynamic approach where user interests vary accordingly to the location, time, and behavior. System complement user sensed on smartphone to suggest user in selecting the better or desired services, photographs or videos dynamically. Since, context are a bit difficult to capture and difficult to describe. For which the fuzzy ontologies and semantic reasoning are used which enriches the description of context.

Graph based recommendation: The graph is generally portioned for recommending videos of latent topic or long tail videos. Apart from which users behavior in social network and can be analyzed by a graph.

As mentioned due to verbose increase of users in numbers, user contexts, user profiles and video contents, the recommendation system requires a lot of computation capacity. Hence, to fix the huge computation requirements, CF algorithms and context-aware algorithms are used on cloud which will enhance and improve the performance and scalability of the proposed system.

2. **Cloud Based Technology**: - As we have discussed the different recommendation systems. Let's discuss the cloud based system for video applications only. Below is the framework of the proposed system is given:

Fig. 1. Proposed Architecture

System majorly includes the two basic parts: recommendation training and real-time recommending. Where recommendation training components collects user contexts, user relationships, and user profiles after which cluster and filter the behavior data on the Cloud to find the recommendation rules. For example, when a user requests new videos, hence real-time recommending components will extend requests to recommendation rules and will return the recommendation.

Lists in accordance with optimized rules. Majorly categorized into four components and the procedures in framework are described below:

1) **Collection of User interest:** Surfing of videos are based on user contexts (time, location, network type), user interests (browsed content, access patterns, and preferred keywords and categories) and friend recommendation/suggestion (reviewed, replied, commented and forwarded). Since, an increase of context types and online users, networking and computing resources will be consumed early. Hence, a lot of issues may rise in order to avoid such issues, collection of contexts are clustered from server side, after which clustering rules are used by application plugging to calculate clusters after clusters are reported to context collectors. With the help of applying this approach, networking and computing load is relieved. Hence, user's social connection and profiles are collected by the collectors at the server side.

2) **Dynamic recommendation of rule generation:** If user content lists are stored in his/her profile due to which storage space gets increase of users and videos, which is resulting to latency in search recommendations lists and unsalable of the system. to avoid such scenarios recommendation rules are extracted from user context clusters and user content clusters. Where rules are composed dynamically.

3) **Optimization real time recommendation:** This is for new user's requests which return the recommendation list to the user who is getting login first time. Its procedure is when a user's requests where recommendation rules are created on basis of request of keywords and user contexts after searched for user favorite as per the rules.

4) **Clustering of user content:** Nowadays user's social connection and profiles are finding on the basis of content. Where social connections are getting retrieved as per the actions on videos shared by users. Apart from which we are also constructing the categories like user profile and server communities are created. Where each of them, content descriptions and content access patterns are getting mapped into tuple.

C. Construction of User Interest

To make a suitable and accurate recommendations for mobile users are completely based on accurate and complete user behaviour models constructed. Since, in today social networking time these are highly influenced by environmental changes and also inheriting the history. Hence, in my system also took example of three kinds of user behaviour : connection of users, access preference and reading interest. Let discuss the above points in detail:

1. Connection between users:

For example in multimedia-sharing websites for example, facebook, flickr, twitter where user assign tags on the re-sources. To analyze the tagging information and users with tagging behaviors show high similarity on some of the specific items. According to the implicit relationship of user to user and user to resource in social networks, the recommendation system can achieve better performance and also lower time cost.

2. **Access Preferences:**

As discussed earlier that user context are very much essential for right amount of service to different users in today's network world. It may concluded that access network types and devices consider length, bit rate and resolution of any of the videos also how much time it takes to access and location of user makes a vital role to be accessed video categorization of the mobile users. As discussed above information like time and location varies and changes person to person, therefore user context needs to be considered on time.

3. **Viewing Interests** Since we can find only a rough guess of a user's interest. But in our case its very important to know the accurate recommendations which depends on user's interest on some content. Users reading interests are generally extracted from his profiles which help to track what video a user viewed. Also we can construct the different concepts like similar access patters, distribution of the topic and preferred entities. Hence, we can construct user's profile majorly in two different concepts: video content and video attributes. Lets discuss below for the same :

1) **Video Attribute:** User may be interested in different attributed like video resolution, video length, video age, video popularity etc. Hence analyzing such historic attributes lists can be retrieved form user profile and we can calculate the probability of user's interest on the clusters of specific video.

2) **Video Content:** In a video we can get some of the keywords like video tags and titles. with the help of which we can create the clusters for same.

D. Proposed Methodology

We have already observed the clustering methods and user interests based on their behaviours. But the recommendations rules needs to be obtained for same where recommender will guide how can we recommend videos to users if a new user is accessing it first time. Below are some of the users which can be defined for related video list play a critical role in the click through rate. same. The collection of related videos in recommendation list and the position of a video in a

As discussed different recommendations rules the execution order can be adjusted before it matches. Here, each rule can be taken as node and different node weight can be calculated by its summary of the rule. Where two rules are connected by an edge and edge is weighted by statics under different constraints of the rules mentioned. As per the rules in above diagram we can also translate into graph given below.

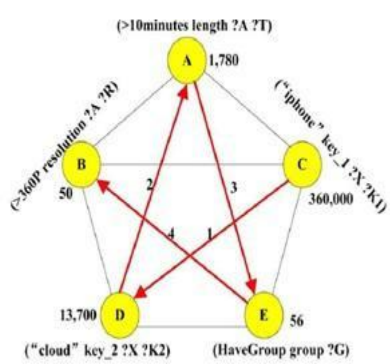

Fig. 2. Rule Execution Reorder

The video recommendation done using algorithm below:

Step 1: Begin with a decision as rating limit (0 to 5) on the value of k= 5 (total number of clusters). The value can be obtained from rating value R given for each video.

Step 2: Put any initial partition that classifies the video rating into k clusters. Assign the videos randomly, or systematically as the following:

Take the first k videos as single-element clusters

Assign each of the remaining (N-k) training sample to the cluster with the nearest centroid.

After each assignment, recomputed the centroid of the gaining cluster.

Step 3: Take each video in sequence and compute its distance from the centroid of each of the clusters. If a video item is not currently in the cluster with the closest centroid, switch this video item to that cluster and update the centroid of the cluster gaining the new video item and the cluster losing the video.

Step 4: Repeat step 3 until convergence is achieved, that is until a pass through the videos causes no new assignment.

IV. PROPOSED RECOMMENDATION MODEL

Algorithm 1: Rank Video BinVideo()

f Input : SearchKey from User U, Record User Rankingg Function BinVideo(SearchKey, Url, ViewCount, Rank) begin Lb : Start Row
Ub : End Row for i=Lb to Ub do
if VideoUrl :: Rank equals 5

fCreate new table Rank5 , insert values such that rank equals 5, Select VideoPk, Url, ViewCount, Rank from TblURL where SearchKey equals SearchName and Rank=5g
End If VideoUrl
if Video Url :: Rank equals 4
f Create new table Rank4 , insert values such that rank equals 4, Select VideoPk, Url, ViewCount, Rank from TblURL where SearchKey equals SearchName and Rank 4g
End If Video Url
if Video Url :: Rank equals 3
f Create new table Rank3 , insert values such that rank equals 3, Select VideoPk, Url, ViewCount, Rank from TblURL where SearchKey equals SearchName and Rank 3g
End If Video Url
if Video Url :: Rank equals 2
Create new table Rank2, insert values such that rank equals 2,
Select VideoPk, Url, ViewCount, Rank from TblURL where

SearchKey equals SearchName and Rank 2

End If Video Url
if Video Url :: Rank equals 1

Create new table Rank1, insert values such that rank equals 1,
Select VideoPk, Url, ViewCount, Rank from TblURL where

SearchKey equals SearchName and Rank 1

End If Video Url
End Loop

VideoRecmd(Tbl h Rank5, Rank4, Rank3, Rank2, Rank1 i)

LTRcmd(TblhRanki) End

Algorithm 2: Recommend Video - VideoRecmd() Function VideoRecmd(h Rank5,Rank4,Rank3,Rank2, Rank1 i)

Input: Tables binned using BinVideo function begin
Create Table Recmd to insert rows Select Top five videoPk ,Url, ViewCount

from Tables h Rank5, Rank4, Rank3, Rank2, Rank1 i whose MAX viewCount

End

Algorithm 3: Update User Rating UpdateRank()

Function
UpdateRank()
Begin
OldCount :viewCount
OldRank : rank
newCount:= OldCount+1;
newRank:=
(OldRank+newRank)/newCount;
h Update newCount, newRank in table TblURL i Bin-Video(SearchKey,Url,newCount,newRank) End

V. MATHEMATICAL MODEL

Let, S=s,e,I,O,F

Here S denotes a video portal with item-based interest clusters and cluster-based item rating in order to recommend newly-generated content items to individual users in real time. S consist of
s = Start of video portal
e = End of video portal
I = Input from users to search a video O = Output of users search

F = Algorithms and functions with computation and time complexity
s = Rq // Request for data
I = D1,D2,D3,: : : ,Dn // List of user profiles

// Perform F on inputs for ranking videos. F = Rank Video Algorithm (BV), Recommended Video (VR), Update User Rating (UR)

IT= CXnet, CXt0, CXdt, CXt1, CXloc)
Input Tuple based on network type (CXnet), device type (CXdt), access time (CXt0), leave time (CXt1), and location (CXloc).

OT= Vres, Vlen, VrateVage, Vcat

Output tuple of video properties Vres represents video resolution, Vlen denotes video length, Vrate denotes video bit rate, Vage is video age, and Vcat represents video categories.
R=Ranking based on rule pairs
$O = Rq^{D}n$

//Collaborative data after applying mentioned algorithms and functions.
e = output as per user request and its authorization
Success condition,
Rq6=NULL, Dn6=NULL
Failure condition, Rq = = NULL, Dn = = NULL

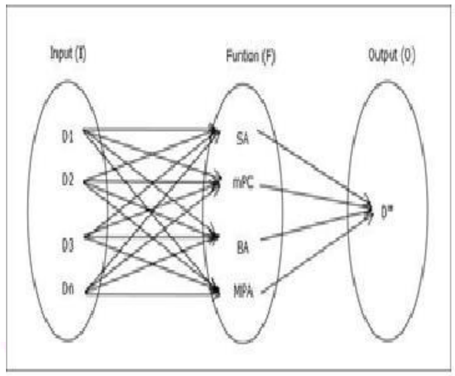

Fig. 3. Mathematical Model

VI. TIME COMPLEXITY

The required time complexity of proposed algorithm for finding the initial centroids is O(nlogn) in both average and worst case, where n is the number of data points. The sorting method used for sorting determines the overall time complexity for finding the initial centroids. Since the proposed enhanced method uses heap sort, its overall time complexity becomes O (nlogn) in both average and worst case. To get the initial clusters the required time complexity is O (nk). Here, some data points stay in the cluster itself and some other data points move to other clusters based on their relative distance from old centroid and the new centroid. If the data point stays in the same cluster then the required complexity is O(1), otherwise O(k). In every iteration the moving of data points to other clusters is decreases. Assuming, until the convergence criteria is met, half the data points move to the other clusters from their present clusters, this requires O(nk/2). Hence the total time complexity for assigning the data points is O(nk), not O(nkl). Therefore the total time complexity of the proposed algorithm becomes O(nlogn). Hence the proposed algorithm has less time complexity compared to the original k-means clustering algorithm.

VII. RESULTS

As per the system we can include into two different parts: first application plugins on mobile terminal, which is implemented on Android platform where it collects user context and takes charge of calculating cluster accordingly context clustering rule. And another part is from server side which is on cloud technology where user context clustering, user group partition, user profile clustering, recommendation rule generation, and real-time recommendation are available.

Where below is the test result of real time recommendation latency which can be categorized in three methods like CF, rule-based with optimization and without optimization. After evaluating the result we come to know that rule based algorithm reduce latency almost 6 times compared to CF.

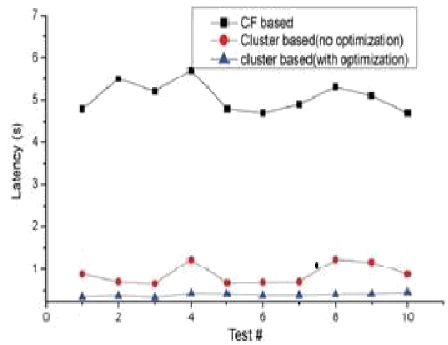

Fig. 4. Comparative Result

VIII. CONCLUSION

In this paper, proposed cloud technology recommendation system only for videos. Here we have analyzed three major kinds of user behaviours like user profiles, user context and interest groups. Also different type of informations like SCA, graph partition and k-means. Majorly graph based rule helps us a lot for real-time recommendation. As per the proposed system which provides higher quality of recommendation.

IX. ACKNOWLEDGMENT

It is a pleasure for me to thank many people who in different ways have supported me in completing this study and contributed to the process of Project. Firstly, I would like to thank my Guide, Prof. P.D. Lambhate for his support during the entire Project work.

X.
Y. REFERENCES

[1] C.-F. Lai, Y.-M. Huang, and H.-C. Chao, DLNA-based multimedia sharing system for OSGI framework with extension to P2P network

[2] K.-D. Chang, C.-Y. Chen, J.-L. Chen, and H.-C. Chao, Challenges to next generation services in IP multimedia subsystem

[3] D. Li, Q. Lv, X. Xie, L. Shang, H. Xia, T. Lu, and N. Gu , Interest based real-time content recommendation in online social communities.

[4] JieBao, Yu Zheng and Mohamed F. Mokbel, A Survey on Recommen-dations in Location-based Social Networks

[5] X.Wu, Y. Zhang, J. Guo, and J. Li,Web video recommendation and long tail

[6] D. Poirier, F. Fessant, and I. Tellier ,Reducing the cold-start problem in content recommendation through opinion classification

[7] M.-H. Kuo, L.-C. Chen, and C.-W. Liang ,Building and evaluating a location-based service recommendation system with a preference adjustment mechanism.

[8] Z.-D. Zhao and M.-S. Shang, User-based collaborative-filtering recom-mendation algorithms on Hadoop

[9] DoshiPoonamPradhumnakumar and Dr. Emmanuel M." Web Pattern Mining using ECLAT" International Journal of Computer Applications (0975 – 8887) Volume 179 – No.8, December 2017.

[10] DoshiPoonamPradhumnakumar and Dr. Emmanuel M." FEATURE EXTRACTION TECHNIQUES USING SEMANTIC BASED CRAWLER FOR SEARCH ENGINE" Jan 2016.

[11] Karan gupta and Poonamlambhate "PROCESSING LINKED MULTIDIMENSIONAL DATA ON THE SEMANTIC WEB" Jan 2016.

5