IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727
PP 59-65
www.iosrjournals.org

Real –Time Video Recommendation System in Hadoop with User Contextual Information

Lakshmi Prakash AV[1], Saini Jacob Soman [2]
[1]*Student, Department of computer Science and Engineering, Sree Naryana Gurukulam College of Engineering, Kerala, India*
[2]*Associate Professor, Department of computer Science and Engineering, Sree Naryana Gurukulam College of Engineering, Kerala, India*

Abstract: *By the end of 2015, 90% of global consumer traffic is being videos that make people hard to obtain their desired content over internet. So the video recommendation system has to take its full potential to provide people with desired items. On behalf of that advanced technologies has to be utilized in existing recommendation system where most of the existing systems suffer from performance as well as scalability problems while dealing with larger datasets. This paper proposed a Hadoop based real–time recommending and recommendation training of a video recommendation system. In real-time recommending section, initially the real-time component accepts the user's request and then process the request and finally returns an individualized recommendation to the user. Here the prior importance is given to the step, user request processing, which is done by recommendation training phase where k-means clustering algorithm on Hadoop platform is executed in the background of system. Meanwhile to provide an individualized recommendation, first of all emulate an actual user behavioral model that means a video portal is developed to feed the data into Hadoop ecosystem to train the real-time recommendation process. This procedure translates the user request into recommendation rules on the basis of performing matching between the user requests that implicit user context with the generated recommendation rule. Besides, if it is a new user request it extends that request in to recommendation rule. Finally system returns an individualized recommendation to the user. Thus this paper can recommend desired services without network overhead so that system can offer massive scalability and apart from that computation complexity can be reduced to a great extent.*

Key Words: *User behavior clustering, MapReduce, K-means algorithm*

I. Introduction

Recent years, the Internet evolving from text to multimedia. As a part of that Video becomes the most popular online service, for an informative and expressive multimedia. Day by day video clips on video portals as well as social network applications keep on increasing. Sometimes the content of video may be similar, duplicate, related, or quite different. Online users usually facing billions of multimedia Webpages, have to experience of hard time in finding their favorites. Some According to video classification, video description tags, or watching history most of the video-sharing websites recommend video lists for end users. However, the final result is that these recommendations are always not consistent with the end user's interests and are not accurate. On behalf of that for almost all existing system initially need to collect user context and then exchange that results that cause heavy network overhead and also the context processing consumes huge computation. Here introduces a novel real-time video recommendation system capable of handling the earlier issues where that make use of Map Reduce programming on a cloud computing platform, namely Hadoop. The main attraction of that can recommend desired services without network overhead and computation issues.

Here in real-time video recommendation system, that make use of Hadoop framework to develop a video portal named "Media Pipe". The recommending system includes two parts: 1) Recommendation training 2) Real-time recommending.

Recommendation training components collect user profiles and then cluster and filter the behavior data on the Hadoop platform to obtain recommendation rules. The real-time recommending component, initially the real-time component accepts the user's request and then process the request and finally returns an individualized recommendation to the user. For that emulate an actual user behavioral model, a video portal is developed to feed input data into the Hadoop ecosystem to train the real -time recommendation process. This procedure translates or extends the new user request into recommendation rules on the basis of performing matching between the new user requests that implicit user context with the generated recommendation rule. The matching procedure translated implicit context rules to searching dimension.

II. RELATED WORK

There are several successful video recommendation algorithms and systems that have been developed and exploited. For almost all of the existing recommendation algorithms, the typical system consists of two essential components. One is recommender that takes charge of user interest identification, user interest recommendation. Other component is collectors that collect user context and activities, content attributes, and updates etc.

Recommendation systems focus on a specific domain. For example, Google News intent a substantial amount of online readers for providing personalized news recommendation services. Amazon make use the recommender system to help users find their desired products. YouTube predicts and recommend videos for users by watching history of users. In general, four categories of algorithms have been exploited by the recommender system: CB recommendation CF-based recommendation, context-aware recommendation, and graph based recommendation.

CB recommendation: It is known as Content-based recommendation. The systems make recommendation based on the similarities of content titles, tags, or descriptions. Some systems find user-interested items based on user's individual reading history in term of content. CB recommender systems are easy to implement. However, in some scenarios, simply representing the user's profile information by a bag of words is not sufficient to capture the exact interests of the user. M. J. Pazzani and D. Billsus [1] define content-based recommendation, which recommends resources based on their content and not on user's rating and opinion. The objects are defined by their associated features of content in the content-based system. The disadvantage of this method is that the resources need to be structural and the taste of users should be described in the features of the content. The content-based systems make recommendation based on the content names, tags, or explanations. Some systems determine user-interested items based on user's individual reading history in term of content. CB recommender systems are very easy to implement. In the content based systems are provided by automatically matching a user's interests with item contents. In content based recommendation very similar items to previous items consumed by the user are recommended which creates a problem of overspecialization.

CF-based recommendation: It is known as Collaborative recommendation. The systems make recommendation based on abundant user transaction histories and content popularity. In the systems, individual user's interests are predicted by a group of similar users. To obtain the content rating and users' similarity, statistics and feedback methods are used. CF systems require enough historical consumption record and feedback. Otherwise, prediction, implicit feedback, or opinion classification methods should be adopted to solve cold-start issues. Z.D. Zhao and M.S. Shang [2] describes a recommender system based on Collaborative Filtering basically predicts a user's interest in some item on the basis of the scores generated and the correlation calculated between the users that use Collaborative Filtering (user based) along with applications of partitioning and clustering of data, thus designing a Recommender System. Finally, out of various scoring functions available, we found Normalized Cosine Vector scoring to be optimum for our dataset. The advantage of Normalized Cosine Vector scoring being that it has a built in dampening effect that eliminates the need of any significance weighting and is preferred over Pearson's correlation as similarity ratings generated by using the vector cosine similarity were found to be more accurate and easy to obtain. Collaborative Filtering (CF) algorithm is a widely used personalized recommendation technique in commercial recommendation systems and many works have been down in this field to improve the performance. However, a big problem of CF is its scalability, i.e., when the volume of the dataset is very large, the computation cost of CF would be very high. Recently, cloud computing have been the focus to overcome the problem of large scale computation task. Cloud computing is the provision of dynamically scalable and often virtualized resources as a service over the Internet.

Context-aware recommendation: The aforementioned systems provide stable recommendation without considering user context information. In fact, user interests vary according to location, time, and emotion. Context-aware recommendation systems complement user context sensed on smartphone and long-time user profile to assist the user in selecting better services, photographs, or videos dynamically. Context is a difficult concept to capture and describe; fuzzy ontologies and semantic reasoning are used to augment and enrich the description of context. P. Pawar and A. Tokmakoff [3] consider a context-based multimedia content management system (MCMS), whose various types of contents are easily gathered from everywhere at any time using mobile phones, and stored in a web server as a multimedia database. There are certain software components for developing location-aware web applications that use multimedia contents stored in the

multimedia database on the web server and also several practical location-aware web applications, e.g., Google Maps based Sight-seeing information system, Google Maps based web Natural Science Dictionary, etc. that are already developed using these components. One of the prospective applications of the proposed framework is Google Maps based Life-log system. Data stored by a lot of users using their mobile phones are regarded as their Life-log data because the data includes location data by GPS, date/time data, other related multimedia data such as picture images, movies, recorded sounds and texts. By analyzing them using any data-mining methods, it is possible to extract activity patterns of the users those are very useful for various web services like recommendation systems.

Graph-based recommendation: Z.Wang, Y. Tan, and M. Zhang [4] describe the Graph –based recommendation built in the system to determine the correlation between filtration objects. The filtration problem turns into a node selection problem on a graph. Incorporating conversion content and contextual information, links on video pages are converted to undirected weighted graph. With the huge increase of user numbers, user contexts, user profiles, and video contents, filtration systems require more and more computation capacity.

G. A. Miller [5] WordNet is a lexical database of English, which groups nouns, verbs, adjectives and adverbs into sets of synonyms (synsets), each expressing a distinct concept.WS4J (WordNet Similarity for Java) provides a pure Java API for several published Semantic Relatedness/Similarity algorithms.Wu-Palmer s computes the similarity between two words by WUP algorithm.JAWJAW (JAva Wrapper for Japanese WorldNet) is a Java API for Japanese WordNet (wn-ja) database (which also contains Princeton's English WordNet v3.0) that offers access to lexical knowledge of a given word
such as hyponym, hyponym, definition, translation (English to Japanese).

Herlihy.M and Luchangco.V [6] describe Hash table based implementation of the Map interface. This implementation provides all of the optional map operations, and permits null values and the null key. (The HashMapclass is roughly equivalent to Hash table, except that it is unsynchronized and permits nulls.) This class makes no guarantees as to the order of the map; in particular, it does not guarantee that the order will remain constant over time. The Hash Map class is the simplest implementation of the Map interface. The Hash Map does not add any additional methods (other than clone) beyond those found in the Map interface. The Hash Map achieves good performance by using a hash to store the key in the Map. The hash allows fast lookup which means that the contain Key( ) method will perform much better than the contains Value( ) method. Any Object used as a key in a Hash Map must implement the hashCode( ) and equals( ) methods.

A.K. Jain and R.C. Dubes [7] explain a K-means algorithm as a widely used clustering algorithm. First, the algorithm randomly selects k initial objects. Each one represents a cluster center. The rest of the objects will be assigned to the nearest cluster, according to their distances to different centers. Then calculate every center again. This operation is repeated until the criterion function converges.

Algorithm description as following:

Input: The number of clusters k and n documents

Output: k clusters

Step1.Randomly select k documents from n documents as the initial cluster centers.

Step2.Calculate the distances of the rest documents to the every center of the clusters, and assign each of the rest documents to the nearest cluster.

Step3. Calculate and adjust each cluster center.

Step4. Iterate Step2 ~ Step3 until the criterion function converge. The program ends.

S. Qin, R. Menezes and M. Silaghi [8] propose a Community-based system. This type of system recommends items based on the preferences of the users friends. This observation, combined with the growing popularity of

open social networks, is generating a rising interest in community-based systems or, as or as they usually referred to, social recommender systems. .This type of recommendation systems models and acquires information about the social relations of the users and the preferences of the user's friends. The recommendation is based on ratings that were provided by the user's friends. In fact these recommendation systems are following the rise of social-networks and enable a simple and comprehensive acquisition of data related to the social relations of the user. Social-network based recommendations are no more accurate than those derived from traditional CF approaches, except in special cases, such as when user ratings of a specific item are highly varied.

R. Burke [9] has proposed Hybrid recommender systems. These systems are based on the combination of the above mentioned techniques. A hybrid system combining techniques A and B tries to use the advantages of A to fix the disadvantages of B. For instance, CF methods suffer from new-item problems, i.e., they cannot recommend items that have no ratings. This does not limit content-based approaches since the prediction for new items is based on their description (features) that are typically easily available. Given two (or more) basic recommendation systems techniques, several ways have been proposed for combining them to create a new hybrid system.

Schafer, J. Ben, Joseph, A. Konstan, John Riedl, [10] proposed a Knowledge-based systems recommend items based on specific domain knowledge about how certain item features meet users' needs and preferences and, ultimately, how the item is useful for the user. Notable knowledge based recommender systems are case-based. Case-based recommenders determine recommendations on the basis of similarity metrics.

III. Problem Statement

Recommendation system have been categorized by four algorithms and exploited them in various systems. The CB recommendation, CF-based recommendation, context-aware recommendation, and graph-based recommendation are the most recognized four algorithms. The aforementioned algorithms are been developed and exploited in the past decade, but a considerable amount of them were constructed and evaluated with small datasets. Furthermore, the volume of web information has greatly increased in the last years, and for that, several recommender systems suffer from performance and scalability problems when dealing with larger datasets. These are all overcome in cloud-based video recommendation system by reducing network overhead and speed up the recommendation computation process. First of all, the recommendation training components collect user contexts and then undergo clustering based on the Map Reduce framework K-means clustering algorithm to obtain recommendation rules. Meanwhile to provide an individualized recommendation, first of all emulate an actual user behavioral model that means a video portal is developed to feed the user request into Hadoop ecosystem to train the real-time recommendation process. This procedure translates the user request into recommendation rules on the basis of performing matching between the user requests that implicit user context with the generated recommendation rule. Besides, if it is a new user request it extends that request in to recommendation rule. Finally system returns an individualized recommendation to the user.

IV. proposed method

The real-time video recommendation system, that make use of Hadoop framework to develop a video portal named "Media Pipe". The system includes two phases: recommendation training and real-time recommending.

In the first phase the recommendation training components collect user profiles and then cluster and filter the behavior data on the Hadoop platform. In real-time recommending section, initially the real-time component accepts the user's request after accepting the request is being processed and then produce an individualized recommendation to the user. Here the prior importance is given to the step, user request processing that is done by recommendation training phase where k-means clustering algorithm on Hadoop platform is executed on the background of system. Meanwhile to provide an individualized recommendation, first of all emulate an actual user behavioral model that means a video portal is developed to feed the data in to Hadoop ecosystem to train the real-time recommendation process. This procedure translates the user request in to recommendation rules on the basis of performing matching between the user requests that implicit user context with the generated recommendation rule. Besides, if it is a new user request it extends that request in to recommendation rule.

4.1 Recommendation Training Phase

Based on the aforementioned description, we propose a novel recommender system for video applications. The framework Recommendation training components is illustrated in Fig 1.

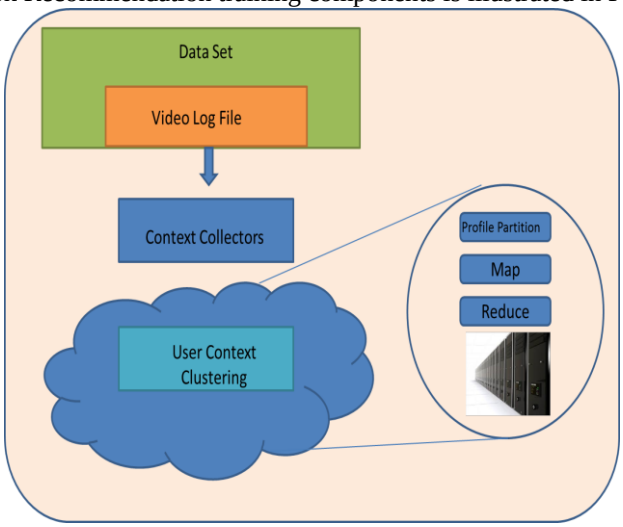

Fig -1: Framework of recommendation training components

Recommendation training components collect video log file with the help of a collector and then cluster and filter the behavior data by means of K-means clustering algorithm to obtain recommendation rules. When a user requests new videos, real-time recommending components will extend requests to recommendation rules and will return the recommendation lists in accordance with optimized rules. The major components and procedures in our framework are described as follows.

User behavior collecting User profiles are being collected where the profile information consists of video attributes such as length, resolution, age and keywords like music, comedy etc. The attribute similarity as well as keyword similarity is being calculated.

User behavior clustering: The component is implemented within the Map Reduce framework. The clustering algorithm is executed on the attribute tuple to obtain user interest clusters and user behavior similarity.

4.2 Real-Time Recommending Phase

Here the real-time recommending component plays the role, where by emulating an actual user behavioral model, a video portal named Media Pipe accepts the user request and then it translate or extend the user input and generate recommendation rule based on clustering algorithm executed on the Hadoop ecosystem. Then based on the generated rule system provides an indivualized recommendation.

Here the real-time recommending component plays the role, where by emulating an actual user behavioral model, a video portal named Media Pipe accepts the user request and then it translate or extend the user input and generate recommendation rule based on clustering algorithm executed on the Hadoop ecosystem. Finally system returns an indivualized recommendation to the user.

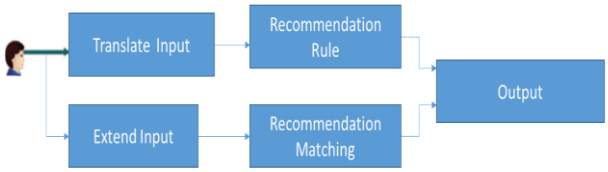

Fig -2: Real-Time Recommending

4.3 Cluster Algorithm

1) Initially user profiles will be stored in HDFS, where the profiles can be stored in different Hadoop nodes.
2) To process the profiles all the mappers are initiated.
3) Then a mapper randomly select k profiles known as central profiles from n profiles as the initial cluster centers and calculate the similarity between the left profiles and the central profiles where both the keyword similarity (i.e. semantic similarity) as well as the attribute similarity has to determine.
4) Then mapper assigns most similar profiles into one cluster and then compute the mean of similarity.
5) Repeat the steps 3 and 4 until the central point remains unchanged or mean of similarity is below a threshold value.
6) After processing each mapper provides intermediate rules with a list of profile attributes.
7) The obtained intermediate rules are then sent in to the reducer to reduce the number of clusters.
8) Then reducer perform same as mapper where central points are chosen and if the profiles are same while comparing then it merged directly.

V. Experimental Analysis

The analysis is mainly done to cross check whether the obtained results are exact or not. It is mainly done on the methodologies which are being used for implementing the whole system. Only by analyzing the methods used we can see whether the proposed method is good or any further modification is needed or not.

5.1 Latency Analysis

The recommendation latency brought by three methods, such as CF,cluster-based algorithm without optimization, and cluster-based algorithm with optimization. The fig 3 shows that cluster -based algorithms reduce latency about six times rather than CF. If the cluster-based algorithms optimized by weighted graph, the latency will be reduced for another 50%.

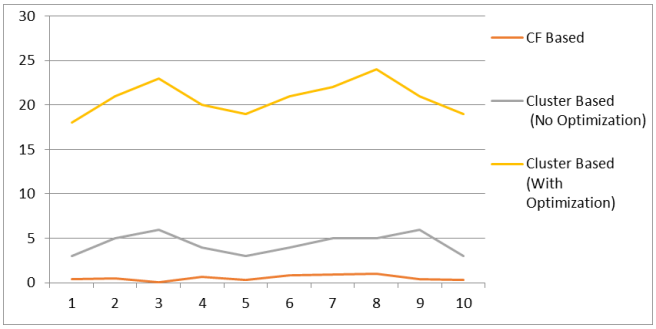

Fig -3: Recommendation latency comparison

VI. Conclusion And Future Work

This paper proposed a recommendation system in hadoop for videos. Based on the MapReduce platform, it mainly analyzed the user behavior. Along with this information, it adopt, K- Means algorithm based on MapReduce framework. This algorithm runs on Hadoop cluster. The results show that the MapReduce framework K-Means clustering algorithm can obtain a higher performance compared to other existing algorithm. Additionally WS4J (WordNet Similarity for Java) provides a pure Java API for several published Semantic Relatedness/Similarity algorithms.Wu-Palmer s computes the similarity between two words by WUP algorithm is used in real-time recommending. Evaluation shows that the proposed system provides higher quality of recommendation with lower training latency and recommending latency.

References

[1]. M. J. Pazzani and D. Billsus, "Content-based recommendatiosystems,"*TheAdaptiveWeb*.Berlin,Germany:Springer-Verlag, 2007, pp. 325–341.
[2]. Z.-D. Zhao and M.-S. Shang, "User-based collaborative-filtering recommendation algorithms on Hadoop," in *Proc. WKDD*, 2010, pp. 478–481
[3]. P. Pawar and A. Tokmakoff, "Ontology-based context-aware service discovery for pervasive environments," in *Proc. IEEE Int. Workshop Service Integr. Pervasive Environ.* Jun. 2006, pp. 1–7.

[4]. Z.Wang, Y. Tan, and M. Zhang, "Graph-based recommendation on social networks," in *Proc. Int. Asia-Pac. APWEB Conf.*, 2010, pp. 116–122.
[5]. G. A. Miller, "WordNet: a lexical database for English," Commun. ACM, vol. 38, no. 11, pp. 39-41, 1995.
[6]. Herlihy, M., Luchangco, V., Moir, M., and Scherer, W. Software transactional memory for dynamic-sized data structures. In Proceedings of the 22nd Annual ACM Symposium on Principles of Distributed Computing (July 2003), pp. 92-101.
[7]. A.K. Jain and R.C. Dubes, Algorithms for Clustering Data. Englewood Cliffs, N.J.: Prentice Hall, 1988.
[8]. S. Qin, R. Menezes and M. Silaghi. A Recommender System for YouTube Based on its Network of Reviewers. In SocialCom '10, 2010.
[9]. R. Burke, "Hybrid recommender systems: Survey and experiments, User Modeling and User-Adapted Interaction," vol. 12, no. 4, 2002, pp. 331-370.
[10]. Schafer, J. Ben, Joseph, A. Konstan, John Riedl, "E-commerce Recommendation Applications", Data Mining and Knowledge Discover, 2001, pp.115-153.