IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 11, November 2015

K-means Clustering with MapReduce Technique

Yaminee S. Patil[1], M. B. Vaidya[2]

Student, Department of Computer Engineering, AVCOE, Sangamner, Maharashtra, India [1]

Assistant Professor, Department of Computer Engineering, AVCOE, Sangamner, Maharashtra, India [2]

Abstract: Clustering analysis is most significant tool for distribution of data. The aim of clustering is to find intrinsic structures in data and categorize them into meaningful subgroups for further study and analysis. In clustering certain assumptions are made about some cluster relationships among the data objects that they are applied on. The process of initiation of cluster formation is based on similarity measure. Unique clusters are formed with the same data set taking help of different notations used in variety of clustering algorithms. K-Means Clustering is one such technique used to provide a structure to unstructured data so that valuable information can be extracted. In this paper we are going to study the implementation of K-Means Clustering Algorithm over a distributed environment using Apache Hadoop. The main focus of the paper is on implementation of the K-Means Algorithm is the design of the Mapper and Reducer routines which has been discussed in the paper. The steps involved in the execution of the K-Means Algorithm has also been described in this paper to serve as a guide for practical implementations.

Keywords: Data mining, clustering analysis, K-means algorithm, Hadoop, MapReduce.

I. INTRODUCTION

Data mining is the process of analysing data from different views and summarizing it into useful information. The data sources for mining purpose are available in variety of forms like, database, data warehouse, the Web, and data which are streamed dynamically in the system. This information is nothing but knowledge which is used to increase the revenue of information. Technically data mining is the process of finding correlations or patterns among the sparse database. As data mining is multidisciplinary field it can be expressed in many ways. Various data mining functionalities are used to recognize patterns in mining task.

Distributed computing is a technique aimed to solve highly complex computational problems over a pool of interconnected systems. Each individual system connected on the network is called a node and the collection of many nodes that form a network is called a cluster.

Cluster analysis is one of the most popular and important topic in data mining. The aim of clustering is to find intrinsic data structures in data, and organize them into meaningful subgroups i.e. clusters for further study and analysis. Each subset is a unique cluster following the principle of maximizing intraclass similarity and minimizing interclass similarity [2]. Different clusters can be formed with same dataset using different clustering method. There are many well established clustering algorithms [1] are present in literature. They can be proposed for very distinct research field, and developed using totally different techniques and approaches. These can be categorized from several orthogonal aspects such as partitioning criteria, separation of clusters, similarity measures used and clustering space.

Apache Hadoop [2] is an open-source software framework written in java for distributed storage and distributed processing of very large data sets on commodity machines. It came into existence from Google's MapReduce and Google File Systems projects. It is a platform that can be used for intense data applications which are processed in a distributed environment. Hadoop framework is designed to compute thousands of petabytes of data. The workload is distributed amongst the computers connected to the network that increases efficiency and overall performance of the system and network. The core Apache Hadoop consists of two parts:

1. Storage Part:- Hadoop Distributed File System (HDFS)
2. Processing Part: - MapReduce Paradigm.

Hadoop also provides a defined file system for organization of processed data which is distributed; scalable and reliable is known as Hadoop Distributed File System (HDFS). Map and Reduce programming paradigm orchestrates the processing by marshalling the distributed servers, running the various task in parallel, managing all communications and data transfers between various parts of system and providing redundancy and fault tolerance. In Map and Reduce the fragmentation of data is the elementary step and this fragmented data is fed into the distributed network for processing. The processed data is then integrated as a whole.

The Hadoop framework [2] [3] takes into account the node failures and is automatically handled by it. This makes hadoop really flexible and a versatile platform for data intensive applications. The answer to growing volumes of data that demand fast and effective retrieval of information lies in engendering the principles of data mining over a distributed environment such as Hadoop.

The remaining of this paper is organized as; in section II, we review literature on cluster analysis. We then brief K-means algorithm and MapReduce technique in section III and section IV respectively. MapReduce algorithm to design MapReduce routines is discussed in section V. And finally we are concluding in section VI.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 11, November 2015

II. CLUSTER ANALYSIS

Data mining is truly multidisciplinary topic which can be defined in many different ways. There are a number of data mining functionalities are used to specify the kinds of patterns to be found in data mining task. These functionalities include characterizations and discrimination, the mining of frequent patterns, associations and correlations, classification and regression; clustering analysis, outlier analysis.

Clustering is one of the most interesting topics in data mining. Clustering has its root in many application areas such as biology, image pattern recognition, security, business intelligence and Web search. The aim of clustering is to find intrinsic structures in data, and organize them into meaningful subgroups for further study and analysis. The basic concept of cluster analysis [1] or clustering is the process of partitioning large data set of objects into small subsets. Each small subset is a unique cluster, such that the objects are clustered together based on the principle of maximizing the intraclass similarity and minimizing interclass similarity. Similarity and dissimilarity are assessed based on the attribute values describing objects and different distance measures. We measures object's similarity and dissimilarity by comparing objects with each other. These measures include distance measures like Euclidean distance, Manhattan distance, supremum distances between two objects of numeric data.[1]

Cluster analysis is a broad subject and hence there are abundant clustering algorithms available to group data sets. Very common methods of clustering involve computing distance, density and interval or a particular statistical distribution. Depending on the requirements and data sets we apply the appropriate clustering algorithm to extract data from them.

On the basis of implementation different clustering algorithm can be grouped together into

- Partitioning Method
  1. K-means
  2. K- medoids
- Hierarchical Method
  1. BIRCH
  2. Chameleon
- Density Based Clustering Method
  1. DBSCAN
  2. OPTICS
- Grid Based Clustering Method
  1. STING
  2. CLIQUE

Advantages of clustering Analysis:

- Improves efficiency of data mining by combining data with similar characteristics so that a generalization can be derived for each cluster.
- Clustering provides meaningful and fastest overview of data repositories.
- Cluster analysis gives a good understanding of the unusual similarities that may occur once the clustering is complete.
- Cluster analysis can be used as a standalone data mining tool to achieve data distribution, or as a pre-processing step for other data mining algorithms operating on the detected clusters.
- Clustering is used in many application areas such as biology, image pattern recognition, security, business intelligence and Web search.

III. K-MEANS ALGORITHM

K-means clustering is used to classify semi structured or unstructured datasets. It is the most frequently used partitional clustering algorithm as it is effective to classify data, handling voluminous data set, simplicity.

The K-means algorithm [1] [4] defines the centroid of a cluster as the mean value of points within a cluster. A centroid-based partitioning technique uses the centroid of a cluster, C_i, to represent that cluster. Conceptually, the centroid of a cluster is its center point. The centroid can be defined in various ways such as by the mean or medoids of the objects (or points) assigned to the cluster. The difference between an object $p \in C_i$ and c_i, the representative of the cluster, is measured by dist (p, ci), where dist (x,y) is the Euclidean distance between two points x and y. the quality of cluster C_i, can be measured by the within cluster variation, which is the sum of squared error between all objects in Ci, and the centroid ci defined as,

$$E = \sum_{i=1}^{k} \sum_{p \in Ci} dist(p, c_i)^2$$

Where E is the sum of the squared error for all objects in the data set; p is the point in space representing a given object; and ci is the centroid of cluster Ci. In other words, for each object in each cluster, the distance from the objects to its cluster is squared, and the distances are summed. The objective function tries to make the resulting K-clusters as unique as possible. The main objective of the algorithm is to obtain a minimal squared difference between the centroid of the cluster and the item in the dataset. The K-means algorithm proceeds as follows:

Algorithm: K-means: the K-mean algorithm for partitioning, where each cluster's centre is represented by the mean value of the objects in the cluster.

Input: K: the number of clusters
D: a data set containing n objects

Output: A set of K cluster

Method:

1. The required number of cluster must be chosen. We will refer to the number of clusters to be 'K'.
2. The next step is to choose distant and distinct centroids for each of the chosen set of K clusters.
3. The third step is to consider each element of the given set and compare its distance to all the centroids of the K clusters. Based on the calculated distance the element is added to the cluster whose centroid is nearest to the element.
4. The cluster centroids are recalculated after each assignment or a set of assignments.
5. This is an iterative method and continuously updated.

DOI 10.17148/IJARCCE.2015.41177

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 11, November 2015

The results may depend on the initial random selection of cluster centres. The time complexity of the k-means algorithm is O (nkt) [9], where n is the total number of objects, k is the number of clusters, and t is the number of iterations. Normally, k<<n and t<<n. Therefore, the method is relatively scalable and efficient in processing large data sets.

Advantages:
1. Simplicity.
2. Understandability.
3. Scalability.

Disadvantages:
1. Sensitiveness to initialization.
2. Cluster size.

IV. MAP-REDUCE TECHNIQUE

Hadoop MapReduce is a software framework for easily writing applications which process vast amount of data like terabytes or even petabytes of data in parallel on large clusters of commodity machines in a reliable, scalable and fault tolerant manner. A MapReduce job usually splits the input dataset into independent chunks which are processed by map tasks in a completely parallel and independent manner. The framework sorts the output s of maps, which are then input to the reduce task. Typically both input and output of the job are stored in file system. Basically the compute node and storage nodes are the same, that is MapReduce and HDFS are running on the same set of nodes. This configuration allows the framework to effectively schedule tasks on the nodes where data is already present.[2]

The MapReduce framework operates exclusively on (key, value) pairs, which is the framework views input to the job as a set of (key, value) pairs and produces a set of (key, value) pairs as output of the job.

Map and Reduce steps are separate and distinct and complete freedom is given to the programmer to design them. Each of the Map and Reduce steps are performed in parallel on pairs of (key, value) data members. Thereby the program is segmented into two distinct and well defined stages namely Map and Reduce. [2] The Map stage involves execution of a function on a given data set in the form of (key, value) and generates the intermediate data set. The generated intermediate data set is then organized for the implementation of the Reduce operation. Data transfer takes place between the Map and Reduce functions. The Reduce function compiles all the data sets bearing the particular key and this process is repeated for all the various key values. The final out put produced by the Reduce call is also a dataset of (key, value) pairs. An important thing to note is that the execution of the Reduce function is possible only after the Mapping process is complete. Each MapReduce Framework has a solo Job Tracker and multiple task trackers. Each node connected to the network has the right to behave as a slave Task Tracker. The issues like division of data to various nodes , task scheduling, node failures, task failure management, communication of nodes, monitoring the task progress is all taken care by the master node.

V. K-MEANS CLUSTERING USING MAP-REDUCE TECHNIQUE

The initial step of designing MapReduce routines K-means algorithm is to define and analyze the input and output of the implementation. Input is given as <key, value> pair, where 'key' is the cluster centroid and 'value' is the serializable implementation of a vector in the dataset. Requirements to implement Map routine and Reduce routine is to have two file one should contain clusters with their centroids and other should have vectors to be clustered. Chosen centroids and the vectors to be clustered are organized in two seprate files is the initial setup for clustering data by K-means algorithm using MapReduce technique of Apache Hadoop. It can be achieved by by following the algorithm to design MapReduce routines for K-means clustering.[8]

The initial set of centroid is stored in the input directory of HDFS prior to Map routine call and they form the 'key' field in the <key,value> pair. The instructions required to compute the distance between the given data set and cluster centroid fed as a <key,value> pair is coded in the Mapper routine. The Mapper computes the distance between the vector value and each of the cluster centeoid mentioned in the cluster set and simultaneously keeping track of the cluster to which the given vector is closest. Once the computation of distances is complete the vector should be assigned to the nearest cluster.

Once Mapper is invoked the given vector is assigned to the cluster that it is closest related to. After the assignment is done the centroid of that particular cluster is recalculated.

The recalculation is done by the Reduce routine and also it restructures the cluster to prevent creations of clusters with extreme sizes i.e. cluster having too less data vectors or a cluster having too many data vectors. Finally, once the centroid of the given cluster is updated, the new set of vectors and clusters is re-written to the disk and is ready for the next iteration.

Algorithm 1: Mapper Design for K-means Clustering

1: **procedure** KmeansMapDesign
2: Load Cluster file
3: f_p = Mapclusterfile
4: Create two list
5: listnew = listold
6: Call read(Mapclusterfile)
7: newf_p = MapCluster()
8: d_v = 0
9: Assign correct centroid
10: read (d_v)
11: calculate centroid
12: d_v = minCenter()
13: Call KmeansReduce()
14: **end procedure** = 0

DOI 10.17148/IJARCCE.2015.41177

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 11, November 2015

Algorithm 2: Reducer Design for K-means Clustering

1: **procedure** KmeansReduceDesign
2: NEW ListofClusters
3: COMBINE resultant clusters from MAP CLASS
4: **if** cluster size too high or too low **then** RESIZE cluster
5: $C_{Max} = \text{findMaxSize(ListofClusters)}$
6: $C_{Min} = \text{findMinSize(ListofClusters)}$
7: **if** $C_{Max} > \frac{1}{20}\text{totalSize}$ **then** Resize(cluster)
8: WRITE cluster FILE to output DIRECTORY
9: **end procedure** = 0

Algorithm 3: Implementation of K-means Function

1: **procedure** Kmeans Function
2: **if** Initial Iteration **then** LOAD cluster file from DIRECTORY
3: **else** READ cluster fie from previous iteration
4: Create new JOB
5: SET MAPPER to map class defined
6: SET REDUCER to reduce class define
8: path for output DIRECTORY
9: SUBMIT JOB
10: **end procedure** = 0

After understanding initial setup of input, output we designed Map and Reduce classes using above algorithms. Next we will see system architecture of our system as shown in following figure.

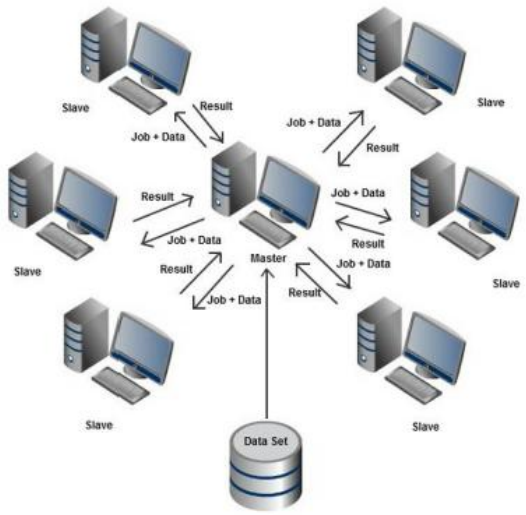

Fig.1 System Architecture

One of the commodity machine from multiple machines which are sharing a private LAN managed by a switch, act as a Master node which supervises data and flow control over all other nodes in the Hadoop cluster[2]. The dataset is stored at the master node. Dataset and Job are distributed amongst the remaining nodes in present network. The Map function is called for the input in form of <key,value> pair. When the assignment is complete, the Reduce function recalculates the centroids and makes the data set ready for the subsequent iterations.

VI. CONCLUSION

Data mining is one of the important tools in information retrieval. The volume of information exchange is growing spectacularly fast which engages huge quantity of data processing. We through this paper have discussed the implementation of K-Means Clustering Algorithm over a distributed network. Not only does this algorithm provide a robust and efficient system for grouping of data with similar characteristics but also reduces the implementation costs of processing such huge volumes of data.

In this paper we have studied significant functionality of data mining that is cluster analysis with K-means algorithm. Theoretical study and empirical examples show that K-means clustering using MapReduce is more suitable for text documents as well as web documents. In this paper we have focussed on K-means clustering algorithm over a distributed environment using Apache Hadoop.

In future we can implement different clustering algorithm using Hadoop framework to reduce the operational resources and increase the speed of operation.

REFERENCES

[1] X. Wu, V. Kumar, J.R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G.J.McLachlan, A. Ng, B. Liu, P.S. Yu, Z.-H. Zhou, M. Steinbach, D.J.Hand, and D. Steinberg, "Top 10 Algorithms in Data Mining,"Knowledge Information Systems, vol. 14, no. 1, pp. 1-37, 2007.
[2] Apache Hadoop. http://hadoop.apache.org/
[3] J. Venner, Pro Hadoop. Apress, June 22, 2009.
[4] T. White, Hadoop: The Definitive Guide. O'Reilly Media, Yahoo! Press, June 5, 2009.
[5] Description if Single Node Cluster Setup at: http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-single-node-cluster/ visited on 21 st January, 2012
[6] Description of Multi Node Cluster Setup at: http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-multi-node-cluster/ visited on 21 st January, 2012
[7] W. Xu, X. Liu, and Y. Gong, "Document Clustering Based on Non-Negative Matrix Factorization," Proc. 26th Ann. Int'l ACM SIGIR Conf. Research and Development in Informaion Retrieval, pp. 267-273,2003
[8] Y. Zhao and G. Karypis, "Empirical and Theoretical Comparisons of Selected Criterion Functions for Document Clustering," Machine Learning, vol. 55, no. 3, pp. 311-331, June 2004.
[9] I. Guyon, U.V. Luxburg, and R.C. Williamson, "Clustering: Science or Art?," Proc. NIPS Workshop Clustering Theory, 2009.
[10] W. Xu, X. Liu, and Y. Gong, "Document Clustering Based on Non-Negative Matrix Factorization," Proc. 26th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 267-273,2003
[11] Yaminee S. Patil, M.B.Vaidya,"A Technical Survey on Cluster Analysis in Data Mining", International Journal of Emerging Technology And Advanced Engineering Website: www.ijetae.com (ISSN 2250 - 2459, Volume 2, Issue 9, September 2012)

DOI 10.17148/IJARCCE.2015.41177