A Survey Paper on K-Means clustering using Hadoop

[1]Meenakshi, [2]Poonam Yadav

[1,2]Gurgaon Institute of Technology & Management, Gurgaon
Email: [1]Mittal_minakshi@yahoo.co.in, [2]poonam.gitm@gmail.com

Abstract: Cluster is a collection of data members having similar characteristics. The process of establishing a relation or deriving information from raw data by performing some operations on the data set like clustering is known as data mining. Data collected in practical scenarios are more often unstructured and semi structured. Hence, there is always a need for analysis of unstructured and semi structured data sets to derive meaningful information. This is where unsupervised algorithms come in to picture to process unstructured or even semi structured data sets. K-Means Clustering is one such technique used to provide a structure to unstructured data so that valuable information can be extracted. This paper discusses the K-Means Clustering Algorithm over a distributed environment using Hadoop(MapReduce). The key to the implementation of the K-Means Algorithm is the design of the Mapper and Reducer routines which has been discussed in the later part of the paper.

Keywords-K-Means Clustering, MapReduce, Hadoop, Data Mining Distributed Computing.

I. DATA MINING

The purpose of the data mining technique is to mine information from a bulky data set and make over it into a reasonable form for supplementary purpose. Data mining is a powerful concept for data analysis and process of discovery interesting pattern from the large amount of data where data is stored in various databases such as data warehouse, World Wide Web, external sources. The goals of data mining are fast retrieval of data or information, knowledge discovery from the databases, to identify hidden patterns and those patterns which are previously not explored, to reduce the level of complexity, time saving etc. It requires accessing and preparing data for a data mining algorithm, mining the data, analyzing results and taking appropriate action. In data mining data can be mined by passing through various phases.

The different phases steps are shown in figure 1.

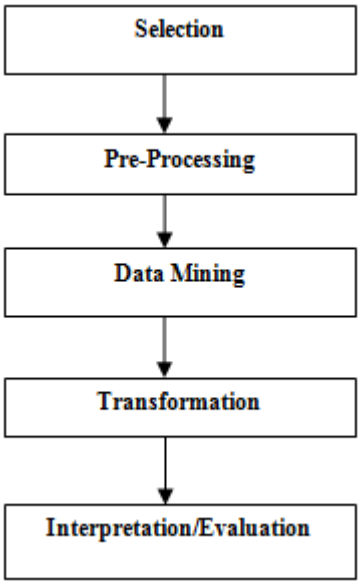

Figure 1: Phases of Data Mining

II. HADOOP

Apache Hadoop is one such open source framework that supports distributed computing. 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. It follows a Map and Reduce programming paradigm where the fragmentation of data is the elementary step and this fragmented data is fed into the distributed network for processing. Hadoop provides a defined file system which is known as Hadoop Distributed File System. 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. This not only reduces the time required for completion of the operation but also reduces the individual system requirements for computation of large volumes of data.

The data-intensity today in any field is growing at a brisk space giving rise to implementation of complex principles of Data Mining to derive meaningful information from the data. The MapReduce structure gives great flexibility and speed to execute a process over a distributed Framework. Unstructured data analysis is one of the most challenging aspects of data

mining that involve implementation of complex algorithms. The Hadoop Framework is designed to compute thousands of petabytes of data. The workload is shared by all the computers connected on the network and hence increase the efficiency and overall performance of the network.

III. CLUSTERING

Clustering is an unsupervised learning in which data are categorized according to their similarities into different groups, and then the groups are labelled. In a cluster Analysis, an automatic process to find similar objects from a database. So a cluster is a collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters. Clustering algorithms are of various types. These are partition-based algorithm(K-Mean), hierarchical-based algorithm, density-based algorithm(DBSCAN) and grid-based algorithm. Its main distinctiveness is the fastest processing time. Depending on the requirements and data sets we apply the appropriate clustering algorithm to extract data from them. In this paper we will discuss K-Means Algorithm using Hadoop in a distributed manner.

The overall process of cluster analysis is shown in fig. 2

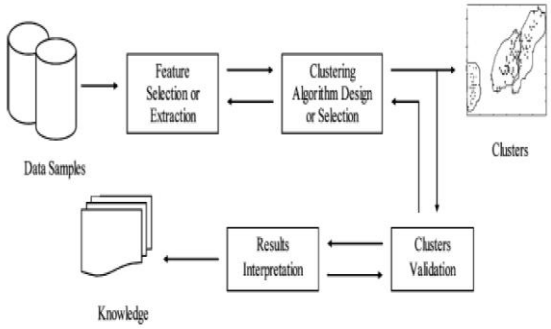

Figure 2: Clustering Process

IV. K-MEANS CLUSTERING

K-means (Macqueen, 1967) is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. K-means clustering is a method of vector quantization, originally from signal processing, that is popular for cluster analysis in data mining. K-Means Clustering is a method used to classify semi structured or unstructured data sets. This is one of the most commonly and effective methods to classify data because of its simplicity and ability to handle voluminous data sets. It accepts the number of clusters and the initial set of centroids as parameters. The distance of each item in the data set is calculated with each of the centroids of the respective cluster. The item is then assigned to the cluster with which the distance of the item is the least. The centroid of the cluster to which the item was assigned is recalculated. One of the most important and commonly used methods for grouping the items of a data set using K-Means Clustering is calculating the distance of the point from the chosen mean. This distance is usually the Euclidean Distance though there are other such distance calculating techniques in existence. This is the most common metric for comparison of points.

Suppose there the two points are defined as

P = (x1(P), x2(P), x3(P) …..) and

Q = (x1(Q), x2(Q), x3(Q) …..).

The distance is calculated by the formula given by

$$d(P,Q) = \sqrt{((x1(P) - x1(Q))^2 + (x2(P) - x2(Q))^2 + \ldots)}$$
$$= \sqrt{\left(\sum_{j=1}^{p} (xj(P) - xj(Q))^2\right)}$$

The next important parameter is the cluster centroid. The point whose coordinates corresponds to the mean of the coordinates of all the points in the cluster. The data set may or better said will have certain items that may not be related to any cluster and hence cannot be classified under them, such points are referred to as outliers and more often than not correspond to the extremes of the data set depending on whether their values or extremely high or low.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.

|xi(j) – cj|2

Where xi is the value of the item and cj is the value of the centroid of the cluster.

The Algorithm is discussed below:

• The required number of cluster must be chosen. We will refer to the number of clusters to be ‘K’.

• The next step is to choose distant and distinct centroids for each of the chosen set of K clusters.

• 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.

• The cluster centroids are re calculated after each assignment or a set of assignments.

• This is an iterative method and continuously updated.

V. MAPREDUCE PARADIGM

MapReduce is a programming paradigm used in Hadoop framework for computation of large datasets. MapReduce implementation splits the huge data into chunks that are independently fed to the nodes so the number and size of each chunk of data is dependent on the number of nodes connected to the network. The Map function is designed by programmer that uses a (key,value) pair for computation. The Map function results in the creation of another set of data in form of (key,value) pair which is known as the intermediate data

set. The programmer also designs a Reduce function that combines value elements of the (key,value) paired intermediate data set having the same intermediate key. Each of the Map and Reduce steps are performed in parallel on pairs of (key,value) data members.

Map: (In_value, In_key) →(Out_key , intermediate_value)

Reduce: (Out_key, Intermediate _value)→ (Out_ value list)

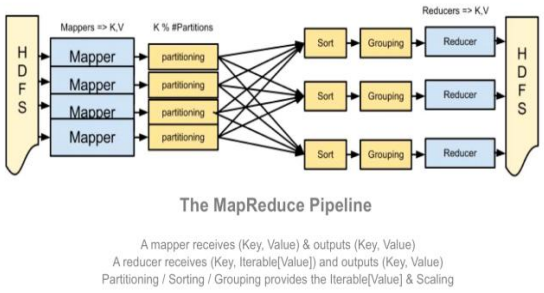

Figure 3: How Map Reduce Works

It means the program is segmented into two distinct and well defined stages namely Map and Reduce. The 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 output 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. The data used as input and output data is stored in the file-system.

VI. K-MEANS CLUSTERING USING HADOOP(MAPREDUCE)

a) How to Map Reduce K-means

• Partition {x1 ,…,xn } into K clusters

– K is predefined

• Initialization

– Specify the initial cluster centers (centroids)

• Iteration until no change

– For each object xi

• Calculate the distances between xi and the K centroids

• (Re)assign xi to the cluster whose centroid is the closest to xi

– Update the cluster centroids based on current assignment.

b) K-MEANS Map/ Reduce Function

- KMeans()

{

Assigncluster()

- For each point p
- Assign p the closest c

Updatecluster ()

- For each cluster
- Update cluster center

}

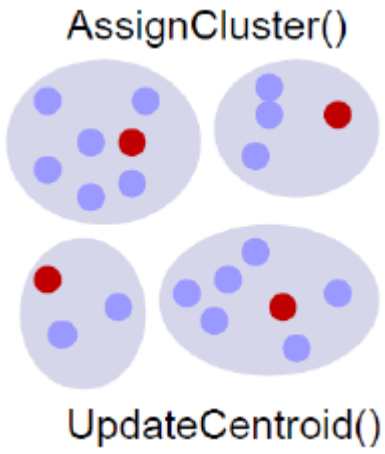

Figure 4: centroid Formation during K-MEAN

c) MapReduce K-means Algorithm

• Driver

– Runs multiple iteration jobs using mapper+combiner+reducer

• Mapper

– Configure: A single file containing cluster centers

– Input: Input data points

– Output: (data id, cluster id)

• Reducer

– Input: (data id, cluster id)

– Output: (cluster id, cluster centroid)

• Combiner

– Input: (data id, cluster id)

– Output: (cluster id, (partial sum, number of points))

The first step in designing the MapReduce routines for K-means is to define and handle the input and output of

the implementation. The input is given as a <key, value> pair , where 'key' is the cluster centre and 'value' is the serializable implementation of vector in the data set. The prerequisite to implement the Map and Reduce function is to have two file one that contains the clusters with their centroids and the other that contains the vectors to be clustered. Once the set of initial set of clusters and chosen centroids is defined and the data vectors that are to be clustered properly organized in two files then the clustering of data using K-Means clustering technique can be accomplished by following the algorithm to design the Map and Reduce routines for K-Means Clustering. The initial set of centers 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 center fed as a <key,value> pair is coded in the Mapper routine. The Mapper is structured in such a way that it computes the distance between the vector value and each of the cluster centers 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.

VII. CONCLUSION

In this Survey paper, we found that there are various means of clustering algorithm to find the problem of arranging clusters .But we present here only K-MEANS algorithm because it is primitive and simplest one for arranging them and unsupervised learning method to solve known clustering issues. its processing time is faster than other kind of clustering algorithm when larger dataset is found. We will do implementation of K-MEAN ALGORITHM using HADOOP as a part of next paper. This will be implemented with big database and using HADOOP.

REFERENCES

[1] Trupti M.Kodinariya(2013),Volume 1, Issue 6**,** Review on determining number of Cluster in k-means.

[2] Amandeep Kaur Mann & Navneet Kaur(2013) ,Review Paper on Clustering Techniques, Volume 13 Issue 5 Version 1.0 .

[3] Brinda Gondaliya(2014) Review Paper on Clustering Techniques, Volume 2 Issue 7, ISSN 2349-4476.

[4] Anjan K Koundinya, Srinath N K, Prajesh P Anchalia(2013) , MapReduce Design of K-Means Clustering Algorithm.

[5] Anjan K Koundinya, Srinath N K, A K Sharma, Kiran Kumar,Madhu M N and Kiran U Shanbagh,Map/Reduce Design anf Implementation of Apriori Algorithm for handling VoluminousData-Sets, Advanced Computing: An International Journal (ACIJ), Vol.3, No.6, November 2012.

[6] www.**tutorials**point.com/**hadoop**.

[7] Clustring_mapreduce.pdf by suny Buffalo.