Repartitioned Optimized K-Mean Centroid Based Partitioned Clustering using MapReduce in Analyzing High Dimensional Big Data

N. Sree Ram, M.H.M. Krishna Prasad, K. Satya Prasad

Abstract*: With the advent of IoT, number of IOT-devices are deployed in the city to acquisition data. These devices acquire enormous data and to analyze such data one need to configure novel hardware to scale up the existing servers and need to develop an application with précised framework. This work recommends an adapted scale out approach in which huge multi-dimensional datasets can be processed using existing commodity hardware. In this approach, Hadoop Distributed File System (HDFS) holds the huge multi-dimensional data to be processed and it can be processed and analyzed by using MapReduce (MR) framework. In the proposed approach, we implemented an optimized repartitioned K-Means centroid based partitioning clustering algorithm using MR framework for Smart City dataset. This dataset contains 10 million objects and each object has six attributes. The results show that the proposed approach is a scalable approach to compute intra cluster density and inter cluster density effectively.

Keywords: Distributed computing, Distributed File System, Hadoop Map Reduce Framework, Inter cluster density, and Re partitioned K-Means.

I. INTRODUCTION

These days IOT getting to be one of the most significant hotspots for information. The interests of information mining strategies are to pick up a ton of data from this profitable source turn out to be progressively essential. Scientific data analysis usually needs a parametric model to characterize given set of data. Data mining algorithms ought to be prepared by the means of suitable computing techniques like distributed computing, map reduce paradigm. Clustering is a vital method in data analysis, and it is using in many domains such as marketing, economics, biology, medicine, weather forecasting, image processing and Web applications. The goal of this is to discover inherent structures in multi-dimensional data and organize them into communicative subsections. The study of "Cluster Analysis-(CA)" is the method of isolating the data objects in dataset into small subsets. Each minor subset is one segment or cluster, such that the data objects are grouped together reliant on the idea of diminishing interclass and exploiting the intraclass similarity. Object's likeness & disparity can be quantifying by comparing objects with one another. Proximities are evaluated based on the dimensions telling objects and various. Proximity measures such as Euclidean distance, Manhattan distance, and supremum distances. There is availability of many clustering algorithms to group the similar data objects in algorithms MAFIA and AMR generate arbitrary shaped clusters by the grid cells to address the said issues.

Big-Data Analytics (BDA) is the process of investigating massive quantity of multi-dimensional data in order to extract concealed patterns, unidentified correlations and other valuable info that can be hand-me-down to make better choices. BDA requires scaling up the hardware platforms turn out to be essential and picking the precise platforms. The selected platform must be able to meet high level of data processing and allow to build BDA solution because the data comes from numerous diverse sources such as smart phones, sensors and soon. Big-Data (BD) refers to massive quantity of data with multi dimensions. At present, because of the employment of cloud-based technologies associated with IOT the dimensions and size of data is more and processing this massive volume of data beyond the current processing capacity. For instance, BD consists of petabytes or exabytes of data consisting billions to trillions of data tuples of a huge number of users from diverse sources for example smart phones, web, banking, medical, business, customers data, social media and soon. Due to this data is from diverse sources usually the data is typically semi structured, unstructured, loosely structured that is often partial and inaccessible.

II. LITERATURE SURVEY

To enhance the novelty in the result of K-mean partitioning clustering algorithm a lot of research happened in earlier days. The data scientists worked on diverse opinion and with diverse ideas. The one methods is that executing the algorithm number of times [1] with arbitrary initial partitions results provide some insight into the quality of ultimate clusters. Forgy's method[2] produces the original cluster by arbitrarily selecting K points first as prototypes and then segmenting the remaining points on the basis of their distance from these centroids Fayyad and Bradley designed an adapted algorithm that employs K-means M times to M arbitrary subsets sampled from the dataset. The

Revised Manuscript Received October 05, 2019
***** Correspondence Author
N.SREERAM*, Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, A.P., India Research Scholar JNTUK, Kakinada A.P, India
Dr.M.H.M. Krishna Prasad, Professor of CSE, Vice-Principal & Coordinator-TEQIP-III,, University College of Engineering Kakinada(A) J.N.T.U. KAKINADA - 533003.
Dr.kK. Satya Prasad, Professor of ECE (Rtd), ,, University College of Engineering Kakinada(A) J.N.T.U. KAKINADA - 533003.

Genetic K-means (GKA) algorithm [3] which integrates a genetic algorithm with K-means to attain a global search and quick intermingling. Global K-means [4] algorithm comprises of a progression of K-means clustering procedures with the number of clusters differing from 1 to K. One weakness of the Global K-Means algorithm is the executing K-Means N times for each value of K, which roots high computational burden for big data sets. The most frequently used method for initialization [5] is selecting K points as centroids in arbitrary fashion from the data set. The primary benefit of this technique is its effortlessness and a chance to cover truly well the arrangement space by complex instatement of the calculation.

A new algorithm named as ISODATA [6], in which estimating K dynamically. For the selection of the correct k value we can order clustering from the possible minimum K to the maximum K by executing k-means several times. The structures produced from the execution of the algorithm then are evaluated using constructed indices and by selecting the best index the anticipated outcome of clustering. [7]. The Cubic Clustering Criterion [8] is the most common method for the identification of number of clusters in k-means.

Cluster Analysis (CA): Clustering is one of the most thought-provoking concepts in knowledge engineering and it is using in many domains such as marketing, economics, biology, medicine, weather forecasting, image processing and web applications. The primary purpose is to detect and organize intrinsic structures in information into communicative sub-categories. The fundamental concept of CA is to group a big information set in tiny sections of objects. Each section is a lonely cluster, so that items are grouped in accordance with the notion of inter-class reduction and the use of intra-class resemblance. The similarity and difference of objects and various range metrics rely on the characteristics of the survey. By contrasting items and each other, we evaluate objects similarity and uniqueness. These metrics include distance metrics such as supremum distances, Manhattan distance, and Euclidean distance. CA [10] is a huge topic and hence there are many clustering algorithms available.

III. PROPOSED WORK

BDA [11] is “the process of investigating big data to learn unseen patterns, obscure correlations and other useful information that can be used in better decision making”. BDA requires scaling up the hardware platforms turn out to be essential and picking the precise platforms. Different big data platforms are accessible. To choose the correct platform for given implementation one should understand all of these platform’s benefits and constraints. The platform chosen must be capable of high information processing and can create a BDA solution, as the information came from myriad sources such as IOT devices, smart phones and soon. Big data is billions of new and constantly updating feed containing location, climate and generates from IOT devices, smart phones, social media and online marketing. Online and start-up companies were the pioneers to seize it. Companies like Facebook, Google and LinkedIn have been constructed from the start around the big data. For example, BD is made up of petabytes or exabytes of data that comprises billions to trillions of data by millions of users from different sources such as smartphones, the web, banking, medicine, enterprises, data from customers, social media and shortly. Because of this information is generally semi-structured, unstructured, and loosely structured from various sources often, partial and inaccessible.

The Lloyd's algorithm, the mostly well known as k-means algorithm, is used to resolve the clustering problem. K-Means is a simple and easy way to define clusters. Because the method is unsupervised, using k-means helps to eliminate subjectively from the analysis. It works as steps mentioned here. First, decide the number of clusters that is k. Then partition the data points into a small number of k groups or clusters. In general, we have n data points xi,i=1...n that have to be partitioned in k clusters. The main task is to allocate each data object to cluster. K-means finds the positions μi,i=1...k of the clusters that diminish the proximity between data points and the cluster. K-means use the amount of the Euclidean distance square, so finding the global minimum is only confidential, potentially resulting in a distinct solution. K-means through MapReduce (MR) can be decompose into the following phases:

Initialization: The given input data set can be split into sub spaces of data. Then the subspace of datasets is shaped into <Key, Value> lists and these <Key, Value> lists input into map function. Select k points as initial centroids arbitrarily from the data sets.

Mapper:

i. The first step in this is compute the distance between each data object in given dataset.

ii. Then allocate each data object to adjacent cluster.

iii. Result the pair <ai, zj> where ai is in the cluster of zj.

Reducer:

i. Take <ai, zj> from Map process. Segregate all the data objects and then output the k numbered clusters and the objects.

ii. Update centroid of cluster by calculating the average of each cluster.

There are some common methods for initializing the position of clusters as follows:

Foggy: Initialize the k observations elected arbitrarily from the given dataset.

Random partition: Allocate a cluster arbitrarily to each observation and compute means as in step 3.

The initial selection of cluster ids is very significant because the algorithm arrives at local minimum.

Algorithm: K-Means

Contribution: Data set D, numbers of clusters k

1. Slave nodes read their part of input data
2. Do until global centroids to the slaves
3. Master node broadcasts the centroids to the slaves
4. Slave nodes assign data instances to the closest centroids
5. Slave nodes compute the new local centroids and local cluster sizes
6. Slaves send local centroids and cluster sizes to the master
7. Master node aggregates local centroids weighted by local cluster sizes into global centroids

Fig:1 K-means algorithm

The reasons why k-means is popular are its time complexity is O(mkl) where m is number of data instances, k is number of centroids and l is number of iterations, its space complexity is O(k+m) and it is order independent.

IV. MAP-REDUCE VERSION OF AN OPTIMIZED K-MEANS

The first step is in developing MR K-Means is to investigate the application inputs and output. The dataset D is fed as <key, value> pair, where "key" is the centroid of cluster and "value" is the serializable implementation of a vector in the dataset D. The requirement to implement Map, reduce routine is to have two diverse files. The first file contains clusters with their centroids values and the second file contains the objects to be clustered. The initial step in K-Means algorithm using MR paradigm is centroids of clusters and the objects to be clustered are stored in two separate files. The algorithm can be complete by the following procedure.The initially selected centroids are stored in the input directory of HDFS former to the Map routine starts and they form the "key" field in the <key, value> pair. The mapper routine includes the instructions for calculating the detachment between the given data objects value and cluster's centroid fed as a pair of <key, value>. It calculates the distance from each cluster centroid between data object value monitor the cluster nearest to which data object are provided. The objects should be allocated to neighboring cluster once the computation have been finished. Once all data objects are allocated to their neighboring clusters each cluster's centroid is recomputed. The reduce routine re-calculates the centroids to stop clusters with dangerous sizes being generated. At the end new set of objects and clusters will be rewritten to memory once a center of the cluster is reviewed

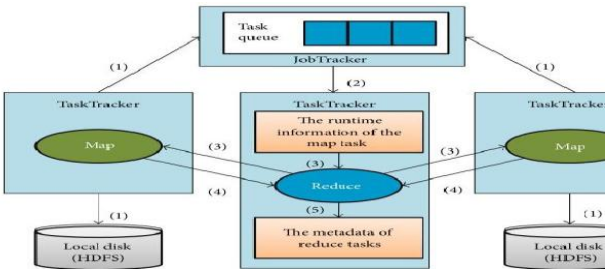

Fig.2: Acquisition of the metadata for reduce

V. REPARTITIONING

The virtual information partitions gathered are partitioned into repartitions with the same amount. After repartitioning the largest partition size can be minimized. Because of repartitioning of virtual partitions, the processing time for maximum partitions will significantly reduce and increase the efficiency of entire reduce phase. It automatically increases the system throughput. In this repartitioning process the virtual partitions generated in map phase are recombined. However, the limitation in physical memory, these virtual partitions will be stored in a local file system. The main difficulty with this repartitioning process lead to the creation of multiplicity of separate virtual partitions in one partition after the balancing process resulting in non-sequential read of disk if it is not properly restricted.

Algorithm 1: Repartitioning algorithm for K-means clustering.
Input: Data set |D| with n attributes a1, a2..., an , and number of partitions K
Result: R: an index of subsequence

```
lb←max (ai)
ub←n
num←0
while lb <ub do
middle←lb + ub – lb /2
foreach ai ∈ A do
sum←sum + a1
if sum >middle then
num++
sum←a1
R←RU i
end
end
if num ≤ K then
ub←middle – 1
end
else if num >K then
lw←middle + 1
end
end
return
```

Fig 3: Repartitioning algorithm for k-means clustering

VI. MAP REDUCE PARADIGM MODEL

Hadoop Map Reduce is programming paradigm which can able to process huge data in parallel fashion with efficiency. This paradigm works in two phases such as Map phase and Reduce phase. Map phase performs filtering and sorting of huge data and Reduce phase performs the consolidation which integrates the outputs and provides the enhanced result. Hadoop HDFS and MR are built with the help of Google file system that provides prepared and acceptable access to data using big clusters of servers. MR is a data processing programming model. Inherent parallels of MR programs allow very large-scale data analysis. MR operates in two stages by splitting the processing model.

Java MapReduce

The map function is depicted by an abstract method map() of mapper class

```
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;
public class MyMapper extends
Mapper<LongWritable,Text,Text,IntWritable>
{
Public void map(LongWritable key, Text value, Context context)
throws IOException, InterruptException
{
}}
```

Fig 4: Map abstract method

The reducer function is similarly define using a Reducer class

```
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Reducer;
public class MyReducer extends
Reducer<LongWritable,Text,Text,IntWritable>
{
Public void reduce(LongWritable key, Text value, Context
context) throws IOException, InterruptException
{
}}
```

Fig 5: Reduce abstract method

Apache YARN has been implemented in Hadoop2 to enhance the application of MR cluster management scheme of Hadoop. Five autonomous entitles exist: The client, which submits the MR job, The YARN resource manager, The YARN node managers, The MR application master, and the distributed file system normally HDFS. In five steps HADOOP performs MR Job: Job submission, Job Initialization, Task Assignment, Task execution and streaming.

Map phase: The proposed work designs the phase of map in master and slave fashion. one of the Master nodes recognizes the input and a complex problem breaks down into smaller sub-problems. It then spread these sub-problems over a slave node in a multi-level tree framework. The slave nodes process the sub problems and return the outcome to the master node.

Reduce phase: Reduce function combines and collects output of all subproblems in the master node and generates the final output. Each map function is associated with a reduces function. The operation mechanism of MR is as follows:

Input: Hadoop based MR framework needs a couple of maps and reduce features that implement the suitable interface or abstract class and should also be defined in the place of input and output and other working parameters. In this stage, the large data in input directory will be divided into several independent data blocks for the Map function of parallel processing.

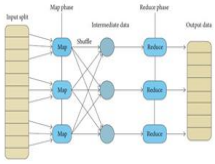

Fig.6. Map Reduce processing model

MapReduce paradigm generate the input as a set of key-value pairs <key, value>. In the Map phase, the paradigm will call the user-defined Map function to process each key-value pair <key, value>, while generating a new batch of middle key value pairs<key, value>.

Shuffle: In order to ensure that the input of Reduce outputted by Map have been sorted, in the Shuffle stage, the paradigm uses HTTP to get associated key-value pairs <key,value> Map outputs for each Reduce; MR paradigm assemblies the input of the Reduce phase according to the key value.

Reduce : This phase will traverse the intermediate data for each unique key and execute user defined reduce function. The input parameter is < key, {a list of values} >, the output is the new key-value pairs< key, value >.

Output : This stage will write the results of the Reduce to the specified output directory location

VII. RESULT AND DISCUSSION

Data set description: The optimized k-mean clustering algorithm is measured with one of the smart city datasets. There are 449 files in the dataset. The polluter proportion of five characteristics is observed at around 17500 in each file.

Experimental setup: we deployed a cluster consists of four nodes in AWS to implement the proposed algorithm.

One Master Node (instance) of type m4, Four slave Nodes (instance) of type m4, Hadoop 2.4.1 and JDK 1.7.

Table 1: Execution Results Of An Optimized Repartitioned K-Means Cluster Algorithm

Dataset	*size*	*Inter-Cluster Density*	*Intra-Cluster Density*
D1	7887974	0.771926	0.572288
D2	6312932	0.783907	0.565958
D3	7099392	0.704998	0.56797
D4	78290	0.689142	0.556309
D5	2368512	0.73014	0.562767
D6	4732842	0.676724	0.611366
D7	1576718	0.740337	0.561887
D8	5522887	0.74722	0.563842
D9	3942470	0.802399	0.567079
D10	3153530	0.748691	0.5684

Evaluation: To quantity the performance of the proposed repartitioned k-means algorithms using Hadoop MapReduce, the algorithm has accomplished on ten myriad samples of dataset.

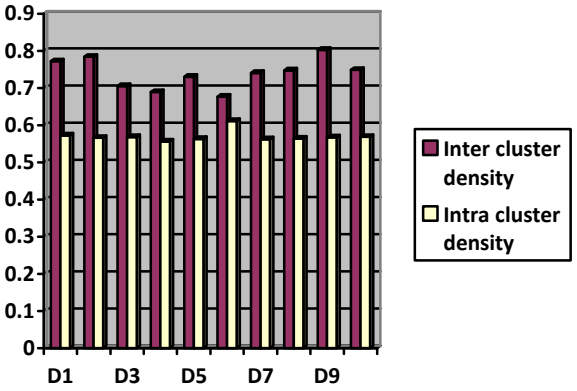

Fig.7: Comparison of Inter cluster density and Intra cluster density

Based on the result after execution of proposed algorithm over dataset indicates that the maximum inter cluster density is 0.802399 for dataset D9 and likewise, the maximum intra-cluster density is 0.611366 for dataset D6 separating data clusters very well.

VIII. CONCLUSION

In this proposed work a scale out approach is designed and implemented to perform clustering on high dimensional big data. HDFS and the Hadoop Map Reduce paradigm has been used to design a framework to perform BDA. By using this framework, an optimized repartitioned k-means clustering algorithm was implemented on Smart City dataset. This dataset contains 10 million objects and each object has six attributes. The proposed approach computes inter cluster density and intra cluster density effectively and compare inter cluster and intra cluster similarity as shown in Figure 3. Hence, this scale out approach is suitable for processing large datasets with the commodity hardware. In relation to which object attribute recognized when a new object is allocated to a cluster are important to comprehend. There may be a greater number of attributes for data scientists to practice in CA, but it is best to decrease as many as possible.

REFERENCES

1. Anil K. Jain and Richard C. Dubes, Michigan State University; Algorithms for Clustering Data:Prentice Hall, Englewood Cliffs, New Jersey 07632. ISBN: 0-13-0222278-X
2. Forgy E (1965) Cluster analysis of multivariate data; efficiency vs. interpretability of classifications. Biometrics, 21: pp 768-780
3. Krishna K, Murty M (1999) Generic K-Means algorithm .IEEE Transactions on systems, man, and cybernetics- part B: Cybernetics, 29(3): pp 433-439
4. Likas A, Vlassis N, Verbeek J (2003) The global K-means clustering algorithm. Pattern recognition, 36(2), pp 451-461
5. Pena JM, Lozano JA, Larranaga P (1999) An empirical comparison of four initialization methods for K-means algorithm. Pattern recognition letters 20: pp 1027-1040
6. Ball G, Hall D (1967) A clustering technique for summarizing multivariate data.Behavioral science, 12: pp 153-155
7. Milligan G, Cooper M (1985) An examination of procedures for determining the number of clusters in a data set. Psychometrika, 50: pp 150-179
8. SAS Institute Inc., SAS technical report A-108 (1983) Cubic clustering criterion. Cary, NC: SAS Institute Inc., 56 pp
9. N.Sreeram and K.Satya Prasad "Efficient map-reduce algorithm for handling very large high dimensional data clustering in heterogeneous multi core environment" Journal of Advanced Research in Dynamical

AUTHORS PROFILE

N.SreeRam, working as Assistant Professor Department of CSE of KLEF Vijayawada, A.P, India , and a research scholar of JNTUK Kakinada A.P, India. Published 8 research papers in various international journals. Life member of Computer Society of India.

Dr MHM Krishna Prasad B.E., M.Tech, MIUR Fellow(U. of Udine, Italy), Ph.D., Professor of CSE, Vice-Principal& Coordinator-TEQIP-III, University College of Engineering Kakinada, J.N.T.U. KAKINADA – 533003 Andhra Pradesh, INDIA State Teacher Awardee from the Govt of Andhra Pradesh

Dr.K.Satya Prasad, worked as a professor of ECE University College of Engineering Kakinada, J.N.T.U. KAKINADA – 533003 Andhra Pradesh, INDIA