Scientific Journal of Impact Factor (SJIF): 4.72

e-ISSN (O): 2348-4470
p-ISSN (P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 4, Issue 10, October -2017

An Optimized repartitioned K-means Cluster algorithm using MapReduce Techniques for Big Data analysis

T.Mohana Priya[1] Dr.A.Saradha[2]

[1]*Research Scholar, Bharathiar University Coimbatore, Tamilnadu,*
[1] *Dr.SNS Rajalakshmi College of Arts and Science, Coimbatore*

[2]*Professor and Head, Department of Computer Science and Engineering,*
Institute of Road and Transport Technology, Erode, Tamilnadu, India

Abstract:- *k-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters fixed apriori. The main idea is to define k centers, one for each cluster. These centers should be placed in a cunning way because of different location causes different result. In this research work, Proposed algorithm will perform better while handling clusters of circularly distributed data points and slightly overlapped clusters.*

Keywords : *K-means algorithm, cluster, big data, hadoop, MapReduce, web logs*

I. INTRODUCTION

With the rapid development of mobile Internet, cloud computing, Internet of things, social network service, and other emerging services, data is growing at an explosive rate recently. How to achieve fast and effective analyses of data and then maximize the data property's benefits has become the focus of attention. The "four Vs" model, variety, volume, velocity, and value, for big data has made traditional methods of data analysis inapplicable. Therefore, new techniques for big data analysis such as distributed or parallelized, feature extraction, and sampling have been widely concerned.

Nowadays internet of things becoming one of the most important sources for data as these data may be used in a lot of application inside smart city which will help to make the life of the human more easy and comfortable. The demand of data mining methods to gain a lot of information from this valuable source becomes more vital. Data mining algorithms should be processed via using suitable computing technique like distributed computing. Distributed computing is a model used to do high computational processing over a set of connected systems. Each individual system interconnected on the network is called a node and the collection of many nodes that form a network is called a cluster. Clustering is an essential method of data analysis through which the original data set can be partitioned into several data subsets according to similarities of data points. It becomes an underlying tool for outlier detection, biology, indexing, and so on. In the context of fuzzy clustering analysis, each object in data set no longer belongs to a single group but possibly belongs to any group.

Apache Hadoop is an open-source software framework that supports data-intensive distributed applications, licensed under the Apache v2 license. It supports the running of applications on large clusters of commodity hardware. Hadoop was derived from Google's MapReduce and Google File System (GFS) papers. The Hadoop framework transparently provides both reliability and data motion to applications. Hadoop implements a computational paradigm named MapReduce, where the application is divided into many small fragments of work, each of which may be executed or re-executed on any node in the cluster. In addition, it provides a distributed file system that stores data on the compute nodes, providing very high aggregate bandwidth across the cluster. Both map/reduce and the distributed file system are designed so that node failures are automatically handled by the framework.

II. RELATED WORK

To get more efficient and effective result of K-mean algorithm there have been a lot of research happened in previous day. All researchers worked on different view and with different idea. Krishna and Murty[4] proposed the genetic K-means(GKA) algorithm which integrate a genetic algorithm with K-means in order to achieve a global search and fast convergence.

Jain and Dubes[1] recommend running the algorithm several times with random initial partitions. The clustering results on these different runs provide some insights into the quality of the ultimate clusters. Forgy's method [2] generates the initial

partition by first randomly selecting K points as prototypes and then separating the remaining points based on their distance from these seeds.

Likas et al. [5] proposed a global K-means algorithm consisting of series of K-means clustering procedures with the number of clusters varying from 1 to K. One disadvantage of the algorithm lies in the requirement for executing K-Means N times for each value of K, which causes high computational burden for large data sets.

Bradley and Fayyad [3] presented a refined algorithm that utilizes K-means M times to M random subsets sampled from the original data. The most common initialization was proposed by Pena, Lozano et al. [6]. This method is selecting randomly K points as centroids from the data set. The main advantage of the method is simplicity and an opportunity to cover rather well the solution space by multiple initialization of the algorithm. Ball and Hall proposed the ISODATA algorithm [7], which is estimating K dynamically. For selection of a proper K, a sequence of clustering structures can be obtained by running K-means several times from the possible minimum Kmin to the maximum Kmax[12].
These structures are then evaluated based on constructed indices and the expected clustering solution is determined by choosing the one with the best index [8]. The popular approach for evaluating the number of clusters in K-means is the Cubic Clustering Criterion [9] used in SAS Enterprise Miner.

III. CLUSTER ANALYSIS

Data mining is interdisciplinary topic which can be defined in many various ways. There are a number of data mining methods are used to determine the types of patterns to be found in data mining task. These methods include discrimination and characterizations, frequent patterns mining, correlations and associations, classification and regression; clustering analysis, outlier analysis. Clustering is one of the most exciting topics in data mining. Clustering used in many application areas such as business intelligence, image pattern recognition, biology, security, and Web search. The objective of clustering is to explore intrinsic structures in data, and arrange them into expressive subgroups. The basic concept of cluster analysis is the process of dividing large data set of objects into small subsets. Each small subset is a single cluster, such that the objects are clustered together depending on the concept of minimizing interclass and maximizing the intraclass similarity. Similarity and dissimilarity are assessed based on the feature values describing objects and various distance measures. We measures object's similarity and dissimilarity by comparing objects with each other. These measures include distance measures such as supremum distances, Manhattan distance, and Euclidean distance, between two objects of numeric data. Cluster analysis is a vast topic and hence there are many clustering algorithms available to group datasets.
On the basis of implementation different clustering algorithm can be grouped together into

Partitioning Method

- K-means
- K- medoids

Hierarchical Method

- Chameleon
- BIRCH

Density Based Clustering Method

- OPTICS
- DBSCAN

Grid Based Clustering Method

- CLIQUE
- STING

Partitioning based Clustering algorithms:
All objects are considered initially as a single cluster. The objects are divided into no of partitions by iteratively locating the points between the partitions. The partitioning algorithms like K-means, K-medoids (PAM, CLARA, CLARANS, and FCM) and K-modes. Partition based algorithms can found clusters of Non convex shapes.

Hierarchical Clustering algorithms:
There are two approaches to perform Hierarchical clustering techniques Agglomerative (top-bottom) and Divisive (bottom-top). In Agglomerative approach, initially one object is selected and successively merges the neighbor objects based on the distance as minimum, maximum and average. The process is continuous until a desired cluster is formed. The Divisive approach deals with set of objects as single cluster and divides the cluster into further clusters until desired no of clusters are formed. BIRCH, CURE, ROCK, Chameleon, Echidna, Wards, SNN, GRIDCLUST, CACTUS are some of Hierarchical clustering algorithms in which clusters of Non convex, Arbitrary Hyper rectangular are formed.

Density based Clustering algorithms:
Data objects are categorized into core points, border points and noise points. All the core points are connected together based on the densities to form cluster. Arbitrary shaped clusters are formed by various clustering algorithms such as DBSCAN, OPTICS, DBCLASD, GDBSCAN, DENCLU and SUBCLU.

Grid based Clustering algorithms:
Grid based algorithm partitions the data set into no number of cells to form a grid structure. Clusters are formed based on the grid structure. To form clusters Grid algorithm uses subspace and hierarchical clustering techniques. STING, CLIQUE, Wave cluster, BANG, OptiGrid, MAFIA, ENCLUS, PROCLUS, ORCLUS, FC and STIRR. Compare to all Clustering algorithms Grid algorithms are very fast processing algorithms. Uniform grid algorithms are not sufficient to form desired clusters. To overcome these problem Adaptive grid algorithms such as MAFIA and AMR Arbitrary shaped clusters are formed by the grid cells.

Model based Clustering algorithms:
Set of data points are connected together based on various strategies like statistical methods, conceptual methods, and robust clustering methods. There are two approaches for model based algorithms one is neural network approach and another one is statistical approach. Algorithms such as EM, COBWEB, CLASSIT, SOM, and SLINK are well known Model based clustering algorithms.

IV. METHODOLOGY

Big Data Analytics
Big data analytics is the process of examining big data to discover hidden patterns, unknown correlations and other useful information that can be used to make better decisions. To perform any kind of analysis on such large and complicated data, scaling up the hardware platforms become necessary and choosing the right platforms becomes a crucial decision to satisfy the user's requirement in fewer amounts of time. There are various big data platforms available with different characteristics. To choose a right platform for specific application one should have knowledge of the advantages and limitations of all these platforms. The platform you choose must be able to cater to increased data processing demands if it is appropriate to build the analytics based solutions on a particular platform.

This data comes from many different sources: The smart phones, the data they generate and consume; sensors embedded into everyday objects, which resulted in billions of new and constantly updating data feed containing location, climate and other information; posts to social media sites, digital photos and videos and purchase transaction records. This data is called big data. The first organizations to grab it were online and startup firms. Firms such as Facebook, Google and LinkedIn are built around big data from the beginning.

"Big Data" refers to data sets too large and complicated containing structured, semi-structured and unstructured data, which is very difficult to handle with traditional software tools. In many organizations, the volume of data is bigger or it moves faster or it exceeds current processing capacity. An example of big data might be Petabytes (1,024 terabytes) or Exabyte's (1,024 petabytes) of data containing billions to trillions of records of millions of various users—all from different sources such as social media, banking, web, mobile, employees and customer's data etc. These types of data are typically loosely structured data that is often incomplete and inaccessible.

K-means algorithm

The Lloyd's algorithm, mostly known as k-means algorithm, is used to solve the k-means clustering problem and works as follows. First, decide the number of clusters *k*. Then:

Clustering is the process of partitioning a group of data points into a small number of clusters. For instance, the items in a supermarket are clustered in categories (butter, cheese and milk are grouped in dairy products). Of course this is a qualitative kind of partitioning. A quantitative approach would be to measure certain features of the products, say percentage of milk and others, and products with high percentage of milk would be grouped together. In general, we have n data points $\mathbf{x}i, i=1...n$ that have to be partitioned in k clusters. The goal is to assign a cluster to each data point. K-means is a clustering method that aims to find the positions $\mu i, i=1...k$ of the clusters that minimize the *distance* from the data points to the cluster. K-means clustering solves

$$\operatorname{argmin}_{\mathbf{c}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in ci} d(\mathbf{x}, \mu i) = \operatorname{argmin}_{\mathbf{c}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in ci} \|\mathbf{x} - \mu i\|_2^2$$

where $\mathbf{c}i$ is the set of points that belong to cluster i. The K-means clustering uses the square of the Euclidean distance $d(\mathbf{x},\mu i)=\|\mathbf{x}-\mu i\|22$. This problem is not trivial (in fact it is NP-hard), so the K-means algorithm only hopes to find the global minimum, possibly getting stuck in a different solution.

Initializing the position of the clusters

It is really up to you! Here are some common methods:

- **Forgy:** set the positions of the k clusters to k observations chosen randomly from the dataset.
- **Random partition:** assign a cluster randomly to each observation and compute means as in step 3.

Since the algorithm stops in a local minimum, the initial position of the clusters is very important.

The pseudo code for k-means clustering algorithm is given below:

Input: Data points D, numbers of clusters k
Step 1: Slaves read their part of data
Step 2: do until global centroids to the slaves
Step 3: Master broadcasts the centroids to the slaves
Step 4: Slaves assign data instances to the closest centroids
Step 5: Slaves compute the new local centroids and local cluster sizes
Step 6: Slaves send local centroids and cluster sizes to the master
Step 7: Master aggregates local centroids weighted by local cluster sizes into global centroids.
Output: Data points with cluster memberships.

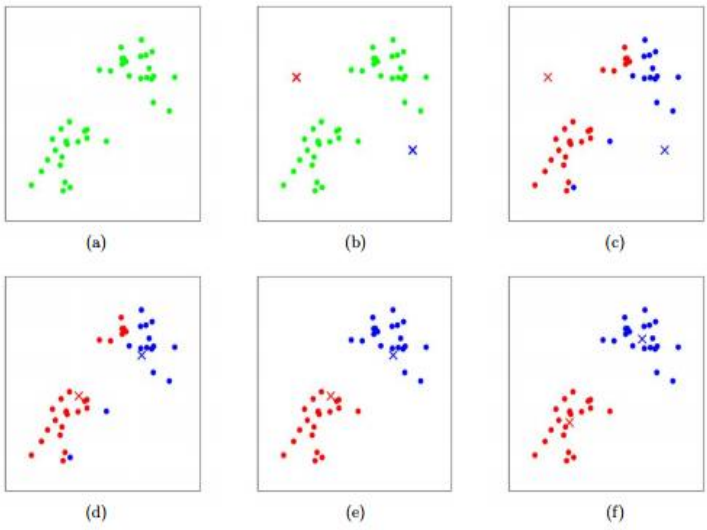

Figure 1: K-means algorithm.

Training examples are shown as dots, and cluster centroids are shown as crosses. (a) Original dataset. (b) Random initial cluster centroids. (c-f) Illustration of running two iterations of k-means. In each iteration, we assign each training example to

the closest cluster centroid (shown by "painting" the training examples the same color as the cluster centroid to which is assigned); then we move each cluster centroid to the mean of the points assigned to it.

In the clustering problem, we are given a training set $x(1),...,x(m)$, and want to group the data into a few cohesive "clusters." Here, we are given feature vectors for each data point $x(i) \in R^n$ as usual; but no labels $y(i)$ (making this an unsupervised learning problem). Our goal is to predict k centroids **and** a label $c(i)$ for each datapoint.

V. An optimized K-MEANS CLUSTERING USING MAP-REDUCE TECHNIQUE

The first step of designing MapReduce code Kmeans algorithm is to express and investigate the input and output of the implementation. Input is given as <key,value> pair, where "key" is the cluster mean and "value" is the serializable implementation of a vector in the dataset. The prerequisite to implement Map routine and Reduce routine is to have two files. The first one should involve clusters with their centroids values and the other one should have objects to be clustered. Chosen of centroids and the objects to be clustered are arranged in two spillited files is the initial step to cluster data by K-means algorithm using MapReduce method of Apache Hadoop.

It can be done by following the algorithm to implement MapReduceroutines for K-means clustering. 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 function calculates the distance between the object value and each of the cluster centroid referred in the cluster set and jointly keeping track of the cluster to which the given object is closest. Once the computation of distances is complete the object should be assigned to the closest cluster.

Once Mapper is invoked, the given object is assigned to the cluster that it is nearest related to. After the assignment of all objects to their associated clusters is done the centroid of each cluster is recomputed.The recalculation is done by the Reduce routine and also it restructures the cluster to avoid generation of clusters with extreme sizes. At the end, once the centroid of the given cluster is revised, the new set of objects and clusters is re-written to the memory and is ready for the next iteration.

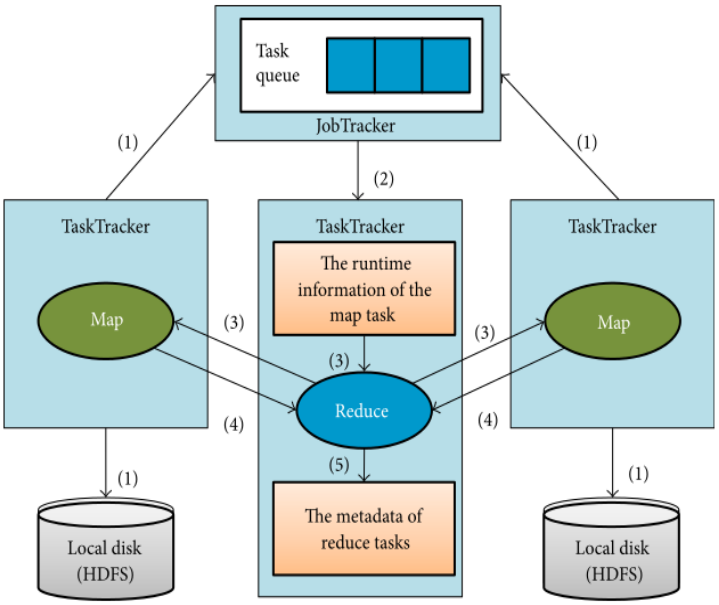

Figure 2: Acquisition of the metadata for reduce tasks.

VI. REPARTITIONING

The repartitioning process divides the collected virtual partitions into new partitions of the same number as reduce tasks. The data size of the biggest partition can be minimized after repartitioning process. It can also reduce the processing time needed for the maximum partition, thereby speeding up the completion of the entire reduce phase and increasing the rate of completed jobs as well as system throughput. As previously analysed, the repartitioning process recombines each virtual partition generated in the map phase. However, due to the limitation of available memory, these virtual partitions must be

written to the local file system. If repartitioning is not restricted, it is likely to lead to a plurality of discrete virtual partitions in one partition following the balancing process, resulting in a non-sequential read of the disk.

Algorithm 1: Repartitioning algorithm.

Data: $A\ a1, a2, \ldots, an\ , K$
Result: R: an index of subsequence
Step 1: $low \leftarrow max\ ai$
Step 2: $high \leftarrow n$
Step 3: $num \leftarrow 0$
Step 4: **while** $low < high$ **do**
Step 5: $mid \leftarrow low + high - low\ /2$
Step 6: **foreach** $ai \in A$ **do**
Step 7: $sum \leftarrow sum$ + a1
Step 8: **if** $sum > mid$ **then**
Step 9: num+ +
Step 10: $sum \leftarrow$a1
Step 11: $R \leftarrow$R$\cup\ i$
Step 12: **end**
Step 13: **end**
Step 14: **if** $num \leq K$ **then**
Step 15: $high \leftarrow$mid − 1
Step 16: **end**
Step 17: **else if** $num > K$ **then**
Step 18: $low \leftarrow$mid + 1
Step 19: **end**
Step 20: **end**
Step 21: return R;

VII. COMPARISON OF CLUSTERING ALGORITHMS

Volume:
It refers to the ability of an algorithm to deal with large amounts of a data. With respect to the Volume property the criteria for clustering algorithms to be considered is a. Size of the data set b. High dimensionality c. Handling Outliers.
Size of the data set: Data set is collection of attributes. The attributes are categorical, nominal, ordinal, interval and ratio. Many clustering algorithms support numerical and categorical data.
High dimensionality: To handle big data as the size of data set increases no of dimensions are also increases. It is the curse of dimensionality.
Outliers: Many clustering algorithms are capable of handle outliers. Noise data cannot be making a group with data points.

Variety:
Variety refers to the ability of a clustering algorithm to handle different types of data sets such as numerical, categorical, nominal and ordinal. A criterion for clustering algorithms is (a) type of data set (b) cluster shape.

Type of data set: The size of the data set is small or big but many of the clustering algorithms support large data sets for big data mining.

Cluster shape: Depends on the data set size and type shape of the cluster formed.

Velocity:
Velocity refers to the computations of clustering algorithm based on the criteria (a) running time complexity of a clustering algorithm.

Time complexity: If the computations of algorithms take very less no then algorithm has less run time. The algorithms the run time calculation done based on Big O notation.

Value:
For a clustering algorithm to process the data accurately and to form a cluster with less computation input parameter are play key role.

VIII. MAPREDUCE PROCESSING MODEL

Hadoop MapReduce processes big data in parallel and provides output with efficient performance. Map-reduce consist of Map function and Reduce function. Map function executes filtering and sorting of large data sets. Reduce function performs the summary operation which combines the result and provides the enhanced output. Hadoop HDFS and Map-Reduce are delineated with the help of Google file system. Google File System (GFS) is developed by Google is a distributed file system that provide organized and adequate access to data using large clusters of commodity servers.

Map phase: The Master node accepts the input and then divides a large problem is into smaller sub-problems. It then distributes these sub-problems among worker nodes in a multi-level tree structure. These sub-problems are then processed by the worker nodes which execute and sent the result back to the master node.

Reduce phase: Reduce function combines the output of all sub problems and collect it in master node and produces final output. Each map function is associated with a reduce function.

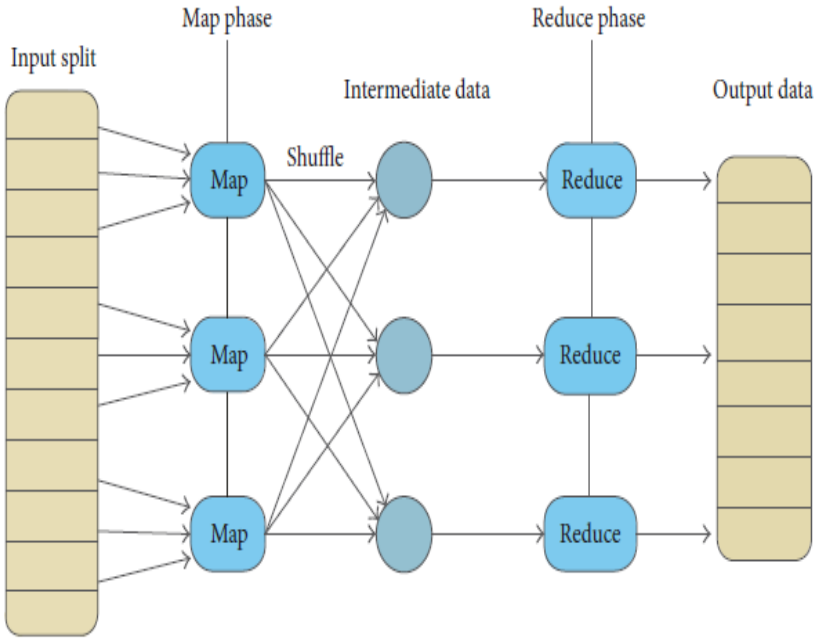

Figure 3: Map Reduce Programming Model

IX. RESULTS AND DISCUSSION

Experimental setup

To implement the k-means algorithm we installed cluster composed of five nodes in aws,

1- One Master Node (instance) of typem4.2xlarge having Ubuntu 14.04, 64 bit.

2- Four slave Nodes (instance) of type m4.large having Ubuntu 14.04, 64 bit.

3- Hadoop 2.4.1

4- JDK 1.7

This distributed environment of four instances in AWS used to implement, perform the optimized repartitioned k-means clustering algorithm and to save the results.

Data set description

To scale optimized repartitioning k-means clustering algorithm one of smart city dataset used . We used the pollution data set which consists of 449 file. Each file contains around 17500 observation of the pollutants ratio of five attributes.

Evaluation

To measure the performance of the scaled k-means algorithms using Hadoop MapReduce, we have executed the algorithms on 10 different samples of data. After execution of the algorithm, we have calculated and measure the inter-cluster and intra-cluster similarity measure.

The inter-cluster distance: distance$d(i,j)$ between two clusters is measured as the distance between the centroids of the clusters.

The intra-cluster distance measured between the all pair of objects within a cluster.

The following table and figure represent the experimental results of K-means algorithm on different data samples where k=3.

Table 1: Execution results of An ptimized repartitioned K-means cluster algorithm

Sample	Sample size	Inter-Cluster Density	Intra-Cluster Density
S1	78290	0.689142	0.556309
S2	1576718	0.740337	0.561887
S3	2368512	0.73014	0.562767
S4	3153530	0.748691	0.5684
S5	3942470	0.802399	0.567079
S6	4732842	0.676724	0.611366
S7	5522887	0.74722	0.563842
S8	6312932	0.783907	0.565958
S9	7099392	0.704998	0.56797
S10	7887974	0.771926	0.572288

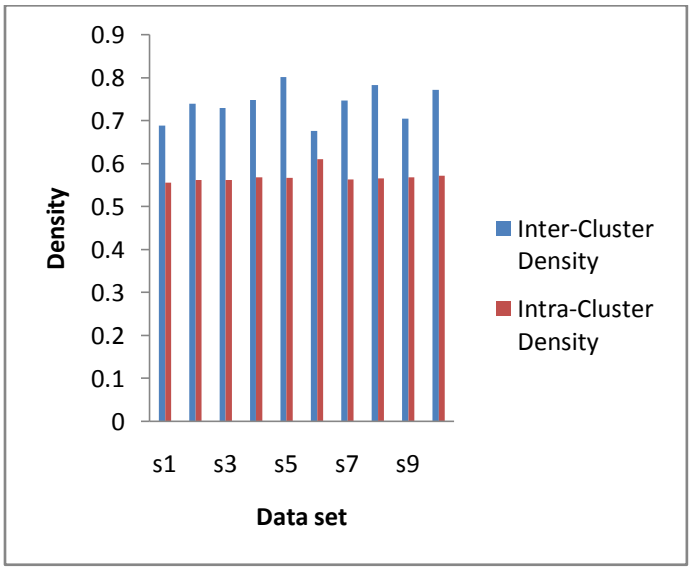

Figure 4 : Comparison between Inter-cluster and Intra Cluster

From the results it is clear the sample s5 shows the maximum inter-cluster density of 0.552399 which indicates well separation of different cluster. Similarly, the inter-cluster density for sample s8 is calculated as 0.533907, separating data clusters very well. Also the results of Intra-cluster density for sample s1 show minimum value, which gives a clear indication of having the similar objects in the same cluster.

X.Conclusion

In this paper Optimized repartitioned k-means clustering algorithm scaled up to be applied to huge dataset which contain around 10 million objects. Each object is a vector of six attributes. Inter and intra cluster measurements computed to find the maximum value of inter-cluster density and the minimum value of intra-cluster measurements. This research work done using Hadoop and MapReduce framework which gives high performance in big data analysis.

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]Bradley P, Fayyad U (1998) Refining initial points for K-means clustering. International conference on machine learning (ICML-98), pp 91-99
[4]Krishna K, Murty M (1999) Generic K-Means algorithm.IEEE Transactions on systems, man, and cybernetics- part B: Cybernetics, 29(3): pp 433-439
[5]Likas A, Vlassis N, Verbeek J (2003) The global K-means clustering algorithm. Pattern recognition, 36(2), pp 451-461
[6]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
[7]Ball G, Hall D (1967) A clustering technique for summarizing multivariate data.Behavioral science, 12: pp 153-155
[8]Milligan G, Cooper M (1985) An examination of procedures for determining the number of clusters in a data set. Psychometrika, 50: pp 150-179
[9] SAS Institute Inc., SAS technical report A-108 (1983) Cubic clustering criterion. Cary, NC: SAS Institute Inc., 56 pp
[10] R.Rajeshkanna, Dr A.Saradha "Cluster Based Load Balancing Techniques to Improve the Lifetime of Mobile Adhoc Networks" Published in International Journal of Trend in Research and Development (IJTRD), ISSN: 2394-9333, Volume-2 | Issue-5 , October 2015.

