International Journal of Research in Advent Technology, Vol.6, No.9, September 2018
E-ISSN: 2321-9637
Available online at www.ijrat.org

Comparative studyon Partition Based Clustering Algorithms

E. Mahima Jane [1] and Dr. E. George Dharma Prakash Raj[2]
1 Asst. Prof., Department of Computer Applications, Madras Christian College, Tambaram – 600 059
2 Asst. Prof., Department of Computer Science and Engineering, Bharathidasan University, Trichy - 620 023.
[1]mahima.jane@gmail.com ,[2]geor9geprakashraj@yahoo.com

Abstract- Clusters are formed by assemblage input data sets in which the objects in the same group are most similar than objects of other groups and the process is called clustering. This paper compares the various enhance k-means partition clustering algorithms for Big Data by proposing new methods to initialize new center values. In order to improve the dependence on the initial values, it proposes various sorting methods. The execution time and the number of iterations are taken into consideration.

Keywords: Clustering, Big Data, Partition, Big Data

1. INTRODUCTION

Clustering is the task of dividing the data points into a number of groups such that data points in the same groups are more alike to other data points in the same group than those in other groups. A Cluster is a set of entities which are similar inside the same cluster which is a powerful mechanism to analyze when the size of the data is huge. In Partitioning-based algorithms, initial groups are specified and reallocated towards a union. These clusters should fulfill the following requirements: (1) each group must contain at least one object, and (2) each object must belong to exactly one group. This paper is structured as follows Section II Related work onpartition based algorithms, Section III discusses the performance of the various enhanced clustering algorithms and Section IV concludes the paper.

2. RELATED WORK

Traditional K Means Algorithm

Yamini[1] k- Means algorithm using Hadoop MapReduce randomly choose centroids and k for the cluster formations.They have performed in distributed environment. Chitra et al [7] and Kavya et al[8] have compared the various clustering algorithms.

The Traditional K means algorithm is given is given below.

1. Step 1: Randomly select k data objects from data set D as initial centers.
2. Step 2: Repeat;
3. Step 3: Calculate the distance between each data object di (1 <= i<=n) and all k clusters C j(1 <= j<=k) and assign data object di to the nearest cluster.
4. Step 4: For each cluster j (1 <= j<=k), recalculate the cluster center.
5. Step 5: Until no change in the center of clusters.

The Enhanced K-Means algorithm is given below in two phases.

AzharRauf[2] In the enhanced K-Means algorithm performs the distance calculation. Using arithmetic mean initial cluster centers are calculated.

Phase-I: To find the initial clusters

Step 1: Find the size of cluster Si (1= i = k) by Floor (n/k). Where n= number of data points Dp (a1, a2, a3 … an) K= number of clusters.

Step 2: Create K number of Arrays Ak

Step 3: Move data points (Dp) from Input Array to Ak until

Si =Floor (n/k).

Step 4: Continue Step 3 until all Dp removed from input array

Step 5: Exit with having k initial clusters.

Phase-II: To find the final clusters

Step 1: Compute the Arithmetic Mean M of allinitial clustersCi

Step 2:Set 1 j k

Step 3:Compute the distance D of all Dp to M of InitialClusters Cj

Step 4:If D of Dp and M is less than or equal tootherdistances of Mi (1 i k) then Dp stay in same cluster

Else Dp having less D is assigned to CorrespondingCi

Step 5:For each cluster Cj (1 j k), Recompute the M

and move Dp until no change in clusters.

SBKMA: Sorting based K- Means Clustering Algorithm using Multi Machine Technique for Big Data[3]is a efficient partition based clustering algorithm which minimizes the execution time and reduces the number of iterations .SBKMA algorithm loads the data into the available nodes. Each nodes are the partition where the data are sorted by the attributes given. All the partitions are merged to form sorted dataset. K clusters are randomly generated. Depending on the size of K the sorted data are partitioned into equal size. Mean of each partition is calculated and taken as initial centroids. Distance is calculated using Euclidean distance. Objects are compared with the initial centroids. Objects are grouped with the nearest cluster. The process of distance calculation and mean are repeated till there is no change in the cluster formation.

SBKMA Algorithm
Step 1:Start
Step 2:Load the dataset into the multiple nodes
Step 3: Generate random value for clusters K
Step4: Divide the dataset D into number of nodes n
Each node is sorted with the pivot element
Step 5: Sorted data Si are divided into K Random generated
Step 6: Mean Mi of every partition is calculated
Step 7: Mean of the datapoints dp is taken as centroids of each cluster
Step 8: Compute the distance between each data point di (1<= i<= n) to all the initial centroids cj (1 <= j <= k).
Step 9: For each data point di, find the nearest centroid cj and assign di to cluster j.
Step 10: Set ClusterNo[i]=j.
Step 11: Set
Clustergroup[i]=d(di, cj).
Step 12: For each data point di,
Compute the distance from the centroid to the nearest cluster
If this distance is less than or equal to the presentcentroid the data point stays in the same cluster
Else
Compute the distance d(di, cj) and recalculate the centroid
End for;
Step 13: Repeat step 9 to 12 till there is no change in the cluster formation.
Step 14: End

SBKMEDA: Sorting based K- Median Clustering Algorithm using Multi Machine Technique for Big Data[4] is an efficient partitionbased clustering algorithm which reduces the execution time even when the data are skewed.SBKMEDA algorithm loads the data in available nodes given. Each partition are sorted by the attributes given. The merged data from all the partitions form the sorted list. The number of cluster K is randomly generated. Depending on the size of the K the sorted data are partitioned into equal size. Median is taken as initial centroids. Distance is calculated using distance formula. Objects are compared with the initial centroids. Objects are grouped with the nearest cluster. Distance calculation and mean calculation for centroids are repeated till there is no change in the cluster formation.

SBKMEDA Algorithm
Step 1:Start
Step 2:Load the dataset into the multiple nodes
Step 3: Generate random value for clusters K
Step4: Divide the dataset D into number of nodes n
Each node is sorted with the pivot element
Step 5: Sorted data Si are divided into K Random generated
Step 6: Median Mi of every partition is calculated
Step 7: Median of the datapoints dp is taken as centroids of each cluster
Step 8: Compute the distance between each data point di (1<= i<= n) to all the initial centroids cj (1 <= j <= k).
Step 9: For each data point di, find the nearest cent*roid cj and assign di to cluster j.
Step 10: Set ClusterNo[i]=j.
Step 11: Set Clustergroup[i]=
d(di, cj).
Step 12: For each data point di,
Compute the distance from the centroid to the nearest cluster
If this distance is less than or equal to the present centroid the data point stays in the same cluster
Else
Compute the distance d(di, cj) and recalculate the centroid
End for;
Step 13: Repeat step 9 to 12 till there is no

change in the cluster formation.
Step 14: End

SBKMMA[5] : Sorting based K Means and Median based Clustering Algorithm using Multi MachineTechnique for Big Datais a clustering based algorithm where the execution time decreases for any type of integer data and reduces the iterations by initializing calculating the centroid values. This algorithm loads the data into the number of nodes given. Total cluster K is randomly generated. Depending on the size of the K the sorted data are partitioned into equal size. Mean and Median of each partition is calculated. When the difference between the mean and median are more median will be taken as centroids else the value of mean will be taken as centroids. Centroids are initialized to the objects which they belong. Distance is calculated using Euclidean distance. Objects are compared with the initial centroids. Objects are grouped with the nearest cluster. Object comparison and distance calculation are repeated till there is no change in the cluster formation.

SBKMMA: Sorting based K Means and Median based Clustering Algorithm using Multi Machine Technique for Big Data
Step 1:Start
Step 2:Load the dataset into the multiple nodes
Step 3: Generate random value for clusters K
Step4: Divide the dataset D into number of nodes n
Each node is sorted with the pivot element
Step 5: Sorted data Si are divided into K Random generated
Step 6: Mean and Median of every partition is calculated.
Step 7: If the value of mean and median differs more
Median will be taken as initial centroids
Else
Mean will be taken as initial centroids
Step 8: Initial data points are assigned to the centroids cj of the clusters they belong.
Step 10: Set ClusterNo[i]=j.
Step 11: Set Clustergroup[i]= d(di, cj).
Step 13: For each data point di,
Compute the distance from the centroid to the nearest cluster
If this distance is less than or equal to the present centroid the data point stays in the same cluster.
Else
Compute the distance d(di, cj) and recalculate the centroid
End for
Step 14: Calculate the distance between the objects to all the centroids and assign it to the nearest
Step 15: Repeat step 10 to 14 till there is no change in the cluster formation.
Step 15: End

3. PERFORMANCE ANALYSIS OF PARTITION BASED CLUSTERING ALGORITHMS

The performance of the clustering algorithms is analyzed using Hadoop MapReduce. The execution time and the numberof iterations are considered.

Table 1: Execution time VS number of nodes

No. of Nodes	K-Means (Seconds)	Enhanced K Means (Seconds)	SBKMA (Seconds)
5 – Nodes	28.2	24.4	20.56
10 – Nodes	23.5	19.33	16.46
15 – Nodes	15.5	11	9.83
20 – Nodes	8	5.9	4.56

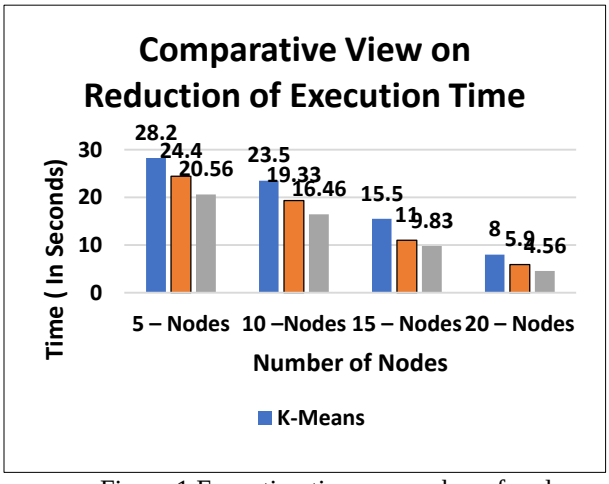

Figure 1 Execution time vs number of nodes

Fig. 1 shows the execution time of three algorithms. From the Table 1 we find the execution time of SBKMA is less when compared to enhanced k-means and traditional k-means. Table shows the execution time using 5nodes 10 nodes 15 and 20 nodes.

Table 2: Execution time VS number of nodes

No. of Nodes	Enhanced K Means (Seconds)	SBKMA (Seconds)	SBKMEDA (Seconds)
5 – Nodes	**24.4**	20.56	18.5
10 – Nodes	**19.33**	16.46	14.3
15 – Nodes	**11**	9.83	8.1
20 – Nodes	**5.9**	4.56	4

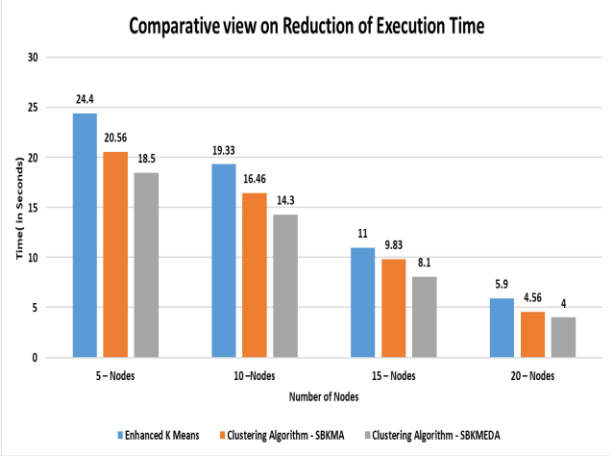

Figure2. Execution time vs number of nodes

Fig. 2 shows the execution time of three algorithms. From the table 2 we find the execution time of SBKMEDA is less when compared to enhanced k-means and SBKMA. Table shows the execution time using 5nodes 10 nodes and 20 nodes. Here median is taken is taken as centroid after sorting which improves the execution time and reduces iterations even when the data items are skewed.

Table 3: Execution time VS number of nodes

No. of Nodes	SBKMA (Seconds)	SBKMEDA (Seconds)	SBKMMA (Seconds)
5 – Nodes	20.56	18.5	15
10 – Nodes	16.46	14.3	13.2
15 – Nodes	9.83	8.1	7.7
20 – Nodes	4.56	4	3.8

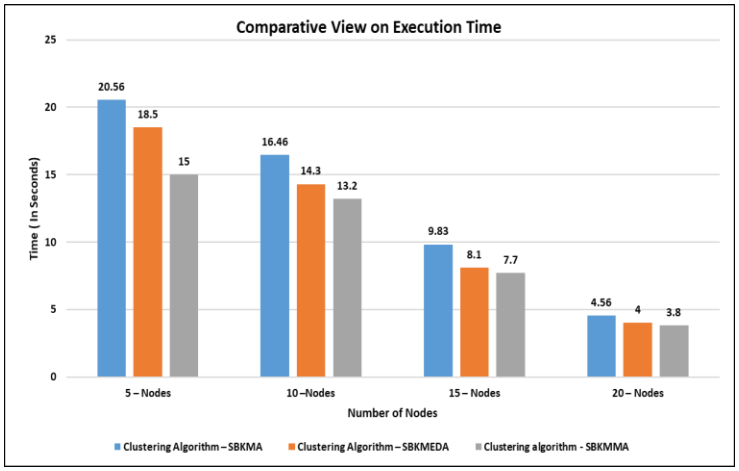

Figure 3: Execution time vs number of nodes

Fig. 3 shows the execution time of three algorithms. From the Table 3 we find the execution time of SBKMMA is less when compared to SBKMA and SBKMEDA. Table shows the execution time using 5nodes 10 nodes and 20 nodes. Here initially centroids are directly assigned to form initial cluster which makes the algorithm to perform faster.

Table 4 : Iterations VS no. of Clusters

No. of Clusters	K-Means (Counts)	Enhanced K Means (Counts)	SBKMA (Counts)
20 Clusters	26	21	16
15 Clusters	19	15	12
10 Clusters	13	11	8
5 Clusters	8	6	4

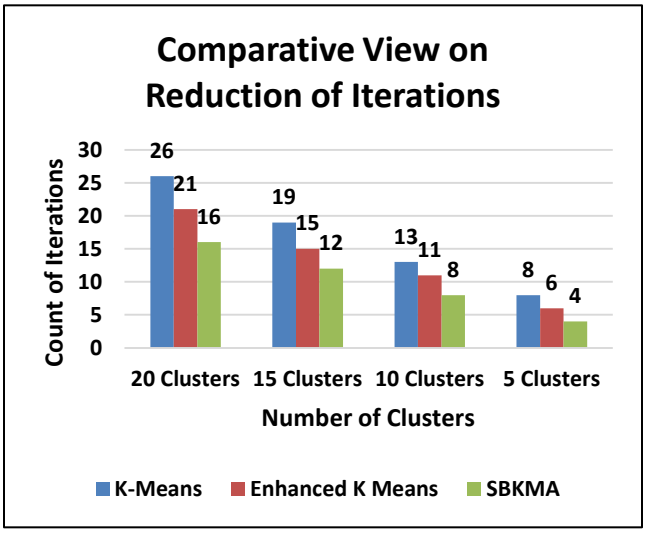

Figure 4 Iterations VS no. of clusters

Figure 4 shows the number of iterations reduces when the number of clusters are decreased. SBKMA uses lessiterations when compared to other two algorithms. Table 4 gives the details about the number of iterations iterationof the various algorithms. Sorting the data in the beginning reduces the repetition in the distance calculation by reducing the number of iterations.

Table5: Iterations VS no. of Clusters

No of Clusters	Enhanced K Means (Counts)	SBKMA (Counts)	SBKMEDA (Counts)
20 Clusters	**21**	**16**	**12**
15 Clusters	15	12	9
10 Clusters	11	8	6
5 Clusters	6	4	3

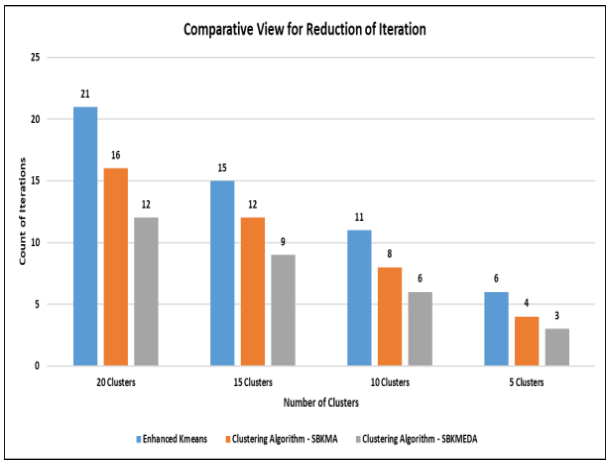

Figure 5 Iterations VS no. of clusters

Figure 5 shows the number of iterations reduces when the number of clusters are decreased. SBKMA uses less iterations when compared to other two algorithms. Table 4 gives the details about the number of iterations iterationof the various algorithms. Sorting the data in the beginning reduces the repetition in the distance calculation by reducing the number of iterations.

Table 6: Iterations VS no. of clusters

No. of Clusters	SBKMA (Counts)	SBKMEDA (Counts)	SBKMMA (Counts)
20 Clusters	16	12	6
15 Clusters	12	9	4
10 Clusters	8	6	3
5 Clusters	4	3	2

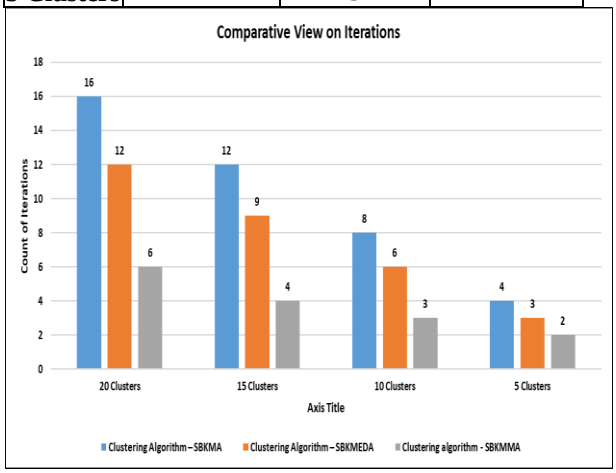

Figure 6 Iterations VS no. of clusters

Fig6 shows the number of iterations reduces when the number of clusters are decreased. SBKMA uses less iterations when compared to other two algorithms. Sorting the data in the beginning reduces the repetition in the distance calculation by reducing the number of iterations in a distributed environment.

4. CONCLUSION

In this paper Partition based ClusteringAlgorithms are compared for Big Data. The ultimate aim of those algorithms is to reduce the execution time by doing it distributed and sorting solves the drawback of iterating data points repeatedly.The performances of traditional K-Means, enhanced K-means, SBKMA, SBKMEDA and SBKMMA are discussed in this paper. The process of initial centroid assignment in multiple nodes reduces the executiontimeincreases the speed of the processing when compared with the previous algorithms. SBKMMAselects initial centroids and assigns centroids to the datapoints to form the initial clusters which makes it fasterin execution time and iterationsthan other existing algorithms.

REFERENCES

[1]Yaminee S. Patil, M. B. Vaidya "K-means Clustering with MapReduce Technique", International Journal of Advanced Research in Computer and Communication Engineering,Nov, (2015)

[2] AzharRauf, Sheeba, SaeedMahfooz, Shah Khusro and HumaJaved " "Enhanced K-Mean Clustering Algorithm to Reduce Number of Iterations and Time Complexity "Middle-East Journal of Scientific Research 12 (7): 959-963, 2012

[3] Mahima Jane and Dr. E. George Dharma Prakash Raj "SBKMA : Sorting based K- Means Clustering Algorithm using Multi Machine Technique for Big Data " in the International Journal of Control Theory and Applications Volume 8 2015pp 2105- 2110

[4] Mahima Jane and Dr. E. George Dharma Prakash Raj "SBKMEDA : Sorting based K- Median Clustering Algorithm using Multi Machine Technique for Big Data " Advances in Intelligent Systems and Computing(Springer) April 2018.

[5] Mahima Jane and Dr. E. George Dharma Prakash RajSBKMMA: Sorting Based K Means and Median Based Clustering Algorithm Using Multi Machine Technique for Big Data. International Journal of Computer (IJC),p. 1-7, Jan. 2018. ISSN 2307-4523.

[6] E. Mahima Jane and E. George Dharma Prakash Raj, "Survey on Partition based Clustering Algorithms in Big Data", International Journal of Computer Sciences and Engineering, Vol.5, Issue.12, pp.323-325, 2017.

[7] K.Chitra and D Maheshwari,"A Comparative study of various clustering algorithms in Data Mining" International Journal of Computer Science and Mobile Computing, Vol.6 Issue.8, August- 2017, pg. 109-115

[8] Kavya D S , Chaitra D Desai "Comparative Analysis of K means Clustering Sequentially Andparallely" International Research Journal Of Engineering And Technology (IRJET) E-ISSN: 2395 -0056VOLUME: 03 ISSUE: 04 | APR-2016.