www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 6 Issue 3 March 2017, Page No. 20647-20650
Index Copernicus value (2015): 58.10 DOI: 10.18535/ijecs/v6i3.46

A Practical Approach for Parallel k-means

Sonal Sharma[1], Preeti Gupta[2], Pooja Parnami[3]
[1] Amity University, Rajasthan, India
Isonal07@gmail.com
[2] Amity University, Rajasthan, India
pgupta@jpr.amity.edu
[3] Amity University, Rajasthan, India
pparnami@jpr.amity.edu

Abstract: ***K-means Clustering is the most popular unsupervised mining technique. The paper explores the standard k-means clustering algorithm and its limitations. Fixing the number of clusters in advance limits the effectiveness of the algorithm. Moreover calculating the distance between each information item and cluster centers, in every single step causes the time complexity of the algorithm to increase. This paper exhibits an improved K-Means algorithm which pre-calculates the optimal number of clusters, dynamically using Dunn's index and achieves high efficiency in terms of execution time, executes the algorithm in a parallel manner using the capabilities of Microsoft's Task Parallel Libraries. Exploratory and comparative results demonstrate that the improved method can effectively enhance the pace of clustering and accuracy when applied on two dimensional raw data consisting of different numbers of records.***

Keywords: *K-means, Improved K-means, Task Parallel library, parallel K-means.*

1. INTRODUCTION

Clustering is an approach that classifies the raw data logically and searches the hidden patterns that may be present in datasets [1]. It is a procedure of collecting data items into disjointed clusters so that the data in the same cluster are similar. The demand for organizing the acute increase in data and taking valuable data from information, makes clustering a procedure broadly applied in numerous application areas for example information retrieval, pattern recognition, artificial intelligence, marketing, data compression, biology, customer relationship management, data mining, , image processing, medicine, machine learning statistics, psychology and so on[2].

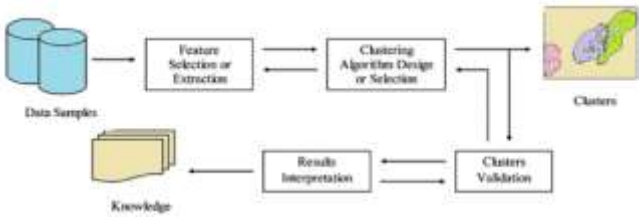

Fig1. Clustering procedure Steps

.

2. LITERATURE SURVEY

2.1 K-Means Algorithm:

K-means is one of the easiest unsupervised learning algorithm that resolve the well known clustering problem. The process follows an easy and uncomplicated method to group a given data set through a specific number of clusters *(k)* fixed apriori. Firstly, *k* is accepted as input , and afterwards information objects which are fitting in with clustering area (including *n* data objects, *n>k*) are alienated into *k* types. Thus, the similitude between same cluster samples is higher. *K* data objects, as novel clustering centers, are arbitrarily selected from clustering.

K-means algorithm steps are as follows:
1) Pick the number of clusters, *k.*
2) Arbitrarily create *k* clusters and find out the cluster centers.
3) Allocate every spot to the closest cluster center.
4) Re-figure the new cluster centers.
5) Reiterate the two preceding steps until some union criterion is met.

2.2 . Shortcomings of K-Means algorithm:

It is observed from the above analysis of algorithm that it has to calculate the distance from each data object to every cluster center in each iteration. This takes more execution time. In traditional k-means algorithm if no points are designated to a cluster during the beginning step, at that time the empty clusters problem occurs.

3. Improved K-Means Algorithm:

3.1. Improved K-Means Algorithm:

To outfit the shortcomings of the existing k means scheme, a solution is proposed that overcomes the limitations of k-means algorithm.The main idea of algorithm is to pre-calculate the optimal no of clusters with the help of Dunn's index and speed up the execution by implement k-means algorithm in parallel environment.

Improved Algorithm:

Input:
K=number of clusters, calculated through Dunn's index
D= A data set containing n items.

Output:

Set of k clusters.

Method:

1. Assign the data point to appropriate cluster center
2. Compute the Intra cluster Maximum distance from all clusters.
3. Calculate the Maximum value from among these distances.
4. Find the Inter cluster Minimum distance from all clusters.
5. Calculate the Minimum value from among these distances.
6. Apply the Dunn's Equation to get the optimum number of clusters

$$V(U) = \min_{1\le i\le c}\left\{\min_{\substack{1\le j\le c\\ j\neq i}}\left\{\frac{\delta(X_i, X_j)}{\max_{1\le k\le c}\{\Delta(X_k)\}}\right\}\right\}$$

7. Update the cluster means,
8. Output the clustering result
9. Apply Task Parallel Library for speed up

Speedup (Sp) = $\frac{Sequential\ time\ for\ scaled-up\ Workload}{Parallel\ time\ for\ scaled\ up}$

3.2 Flow chart:

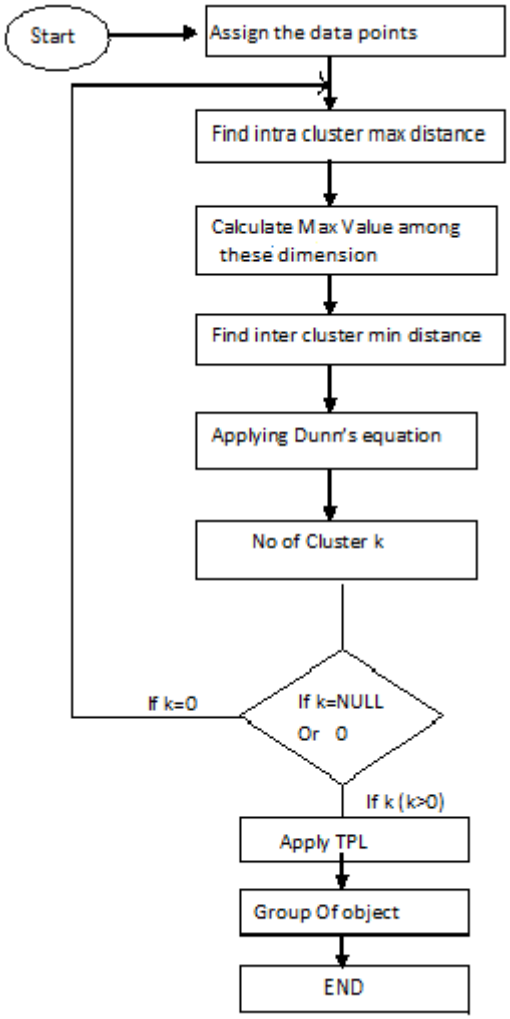

Fig 3.Steps of Improved K-Means

3.3. *Dunn's Index:*

This index identifies sets of clusters that are well divided.

Dunn's index is defined as:

$$V(U) = \min_{1\le i\le c}\left\{\min_{\substack{1\le j\le c\\ j\neq i}}\left\{\frac{\delta(X_i, X_j)}{\max_{1\le k\le c}\{\Delta(X_k)\}}\right\}\right\}$$

Where $\delta(X_{i,}$ describes the distance between cluster and (inter-cluster distance), Δ signifies the intra-cluster distance of cluster and K is the number of clusters in partition U. The main goal of Dunn's index is to maximize inter-cluster distances while minimizing intra cluster distances. Thus large values of Dunn's index correspond to good clusters. Therefore, the number of clusters that maximizes the index is obtained as the finest number of clusters. The study finds its base in work of Bolshavoka [4] and Azuaje [5].

3.4. Parallelization of K-Means:

. The Task Parallel Library (TPL) is based on the concept of a task, which represents an asynchronous operation. In some ways, a task resembles a thread or Thread Pool work item, but at a higher level of abstraction. The term task parallelism refers to one or more independent tasks running concurrently. This library is of .NET environment.

4. RESULTS & ANALYSIS

All the programs are written in Microsoft Visual studio.net(C#). Different machine architectures may differ greatly on the total runtime for the same algorithms. The run time used here means the total execution time that is, the period between input and output, instead of CPU time measured in the experiments in some literature. To illustrate the numerical behavior of the modified k-mean algorithm and to compare it with the standard k-mean algorithm the algorithms are implemented on the same data set. In this paper, the most representative algorithms K-Means and proposed algorithm, modified K-Mean were examined and analyzed based on their basic approach for large data set.

The best algorithm in each category was found out based on their performance. Comparison between K-Means and modified K-Mean algorithm with numbers of records and execution time (in milliseconds) is shown in the Table I.

1. Following results were found for serial and parallel K means while having cluster size of 9, on different size of datasets.

Data (no. of records)	K-Means	Parallel K-Means	SpeedUP
400	985	865	13
1200	1854	1358	36

DOI: 10.18535/ijecs/v6i3.46

2300	2640	1743	51

Fig 1: Comparison b/w Serial k-means versus Parallel k-means, *k*=9 Clusters...

2. These results came out from the analysis between the Serial and Parallel K-Means with Dunn's index. Here operations are performed on different size of datasets and in Dynamic K-Means clusters size is calculated by using Dunn's Index.

Data	K-Means	Parallel K-Means	Speed UP
100	390	312	25
400	1097	803	36
1200	1420	1150	23
1800	1710	1340	27
2300	2551	1760	44

Fig 3: Comparison b/w Serial k-means versus Parallel k-means, Calculated by Dunn's index

3. Following is the analysis between the Serial and Parallel K-Means with Dunn's index. Here operations are performed on different size of datasets and in Dynamic K-Means clusters size is calculated by using Dunn's Index. This figure shows the speed up of parallel operation over serial operation

Fig 4: Speedup Ratio between K-means & parallel k-means.

4. These are the results for the K-Means Algorithm that makes the clusters according their similarity and having minimum distance.

Data	Clusters
200	6

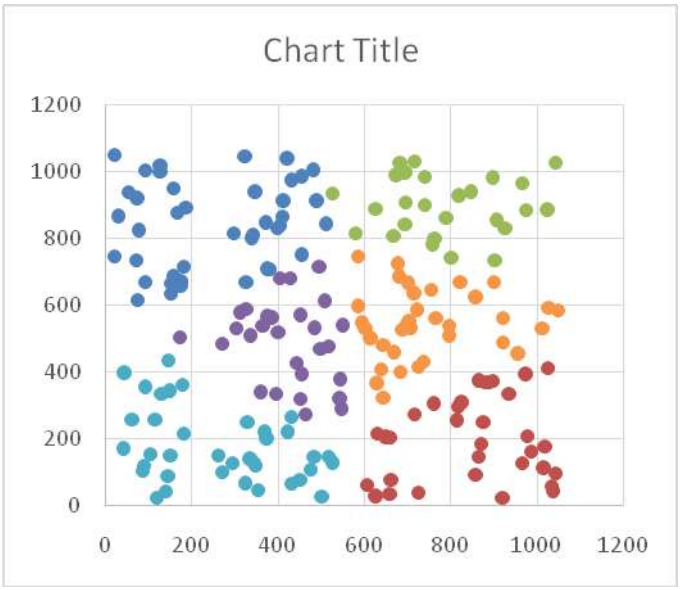

Fig.6: Resulting 6 Clusters for *k*=200.

5. CONCLUSION

This paper proposed an improved data clustering method for the anonymous data set. The algorithm works fine for the unknown data set with improved results than traditional K-means clustering. The k-means algorithm is well known for its ease and the alteration is done in the proposed method with maintenance of simplicity. The traditional K-means algorithm obtains number of clusters (K) as input from the user. The major problem in traditional K-means algorithm is fixing the number of clusters in advance.

The results shows that the proposed approach has overcome this problem by finding the optimal number of clusters on the run with the help of Dunn's index, and use of parallel libraries enables the algorithm to perform better than conventional K-means algorithm in all scenarios with small or large datasets.

6. FUTURE WORK

Future work can focus on how to reduce the time complexity without compromising cluster quality and optimality. More experiments can be conducted with natural datasets with different features. use some more powerful parallel programming models like Intel's Cilkplus and OpenMP to obtain reduced execution time

REFERENCES

1. International Conference on Information and Computer Networks, "Dynamic Clustering of Data with Modified K-Means Algorithm" Ahamed Shafeeq B M and Hareesha K S 2012 (ICICN 2012) IPCSIT vol. 27 (2012) © (2012) IACSIT Press, Singapore

2. "Top 10 algorithms in data mining", Knowledge and Information Systems, January 2008, Volume 14, Issue 1, pp 1-37, 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, D. Steinberg,.
3. "Dynamic load balancing on GPU clusters for large-scale K-Means clustering, " E. Kijsipongse, S. U-ruekolan2012 IEEE International Joint Conference on Computer Science and Software Engineering (JCSSE), vol., no., pp.346,350, May 30 2012-June 1 2012

4. Likas,A., Vlassis, M. & Verbeek, J. (2003), The global k-means clustering algorithm, Pattern Recognition, 36, 451-461

5. N. Bolshakova, and F. Azuaje, "Cluster validation techniques for genome expression data," Signal Processing, vol.83, pp.825-833, April 2003.

6. F. Azuaje, "A cluster validity framework for genome expression data, "Bioinformatics, vol.18, pp.319–320, February 2002
7. Third International Symposium on Intelligent Information Technology and Security Informatics "An Improved k-means Clustering Algorithm", Shi Na, Liu Xumin, Guan yong College of Information Engineering, Capital Normal University CNU Beijing, China

8. R. Parikh,"Accelerating quicksort on the intel Pentium 4 processor with hyper–threading technology",Software Community Intel, October 2007