Prof P M Chawan, Saurabh R Bhonde, Shirish Patil / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue 2,Mar-Apr 2012, pp.1378-1382

Improvement of K-Means clustering Algorithm

Prof P M Chawan VJTI, Mumbai
Saurabh R Bhonde VJTI, Mumbai
Shirish Patil VJTI, Mumbai

ABSTRACT

By the help of large storage capacities of current computer systems, datasets of companies has expanded dramatically in recent years. Rapid growth of current companies' databases has raised the need of faster data mining algorithms as time is very critical for those companies.
Large amounts of datasets have historical data about the transactions of companies which hold valuable hidden patterns which can provide competitive advantage to them.

In this project, K-means data mining algorithm has been proposed to be improved in performance in order to cluster large datasets in shorter time. Algorithm is decided to be improved by using parallelization. Parallel version of the K-means algorithm has been designed and implemented by using C language. For the parallelisation, MPI (Message Passing Interface) library has been used.

Keywords: K-means, parallel algorithm, MPI, intracluster.

1. Introduction

Parallel version of the K-means algorithm has been designed and implemented in this project for the purpose of improvement of K-means algorithm in execution time. Serial (Classical K-means) version of the algorithm has also been implemented for the purpose of comparison with parallel the version in time. Both implementations have been tested on the same environment and results have been discussed.

As K-means is a clustering algorithm which is a type of data mining algorithm, data mining and clustering have also been examined in the project.

KDD (Knowledge Discovery in Databases) has also been discussed, because data mining is a step of it. After addressing where K-means stands, details of serial and proposed parallel K-means algorithms has been presented. Before examining parallel K-means algorithm, parallelisation concept of algorithms has been introduced in order to prepare a background for the details of parallel K-means algorithm.

2. Knowledge discovery in databases and data mining

KDD refers to the overall process of discovering useful knowledge from databases.

KDD consist of several steps. Data mining refers to a particular one of those steps of overall KDD process. Data mining is the application of specific algorithms for extracting patterns, which then will be interpreted and evaluated to produce knowledge. Main aspect of this Master's Thesis Project is the data mining itself, not the whole KDD process. Therefore, data mining will be examined in more detail then the overall KDD process.

In addition to data mining step, KDD process also has data selection, preprocessing, transformation and interpretation steps. Composition of these steps constitutes the KDD process. In order to understand what data mining is and address where it stands. overall representation of KDD process, which also
includes data mining step, can be seen.

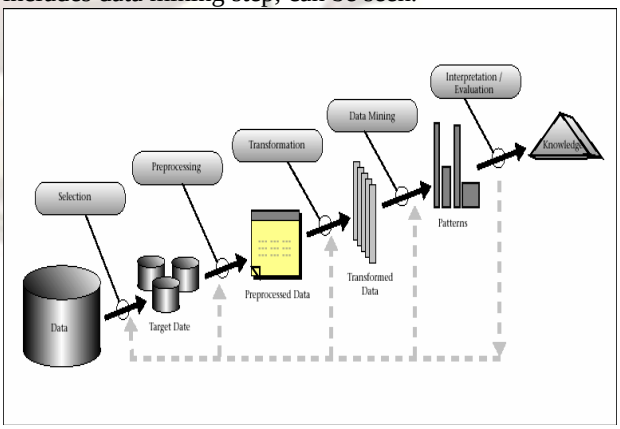


Figure 2.1 Overall Representation of KDD Process

KDD process consists of Data Selection, Preprocessing, Transformation, Data Mining and Interpretation / Evaluation steps. First step of KDD process is to create a target dataset on which KDD will be performed. This is to select an application dataset or a subset of it. Second step of KDD is the cleaning and preprocessing. Basically, removal of noise in data is performed and strategies for handling missing data fields are developed in this step. As a third step, transformation of preprocessed data is performed. This is to find useful features of data which defines the data according to the needs of data mining algorithm. Next step is the data mining itself, which uses transformed data and produces patterns and relationships. Final one is the interpretation and evaluation step which is to comment on mined patterns in order to develop knowledge. In this step, return back to each of other steps may be performed for further iterations.

2.1 Data Mining

As mentioned above, data mining is the task of discovering hidden patterns and relationships in databases which are prior to knowledge production. In companies' large databases, there are lots of hidden patterns of strategic importance. Data mining is the only method of digging these databases and finding these valuable patterns. Without data mining, it is impossible to examine such large databases and produce valuable information. Data mining is very critical for companies in order to produce strategic information by using their historical data. By using data mining, companies can control their costs and increase revenue [Palace 1996]. Currently, data mining is being used in wide variety of business areas for lots of purposes. Most organizations use data mining in order to manage their customer life cycle such as acquiring new customers, increasing revenue of existing ones and retaining good customers. Data mining is strongly related and supported with some other data processing and statistical works.

2.2 Types of Data Mining Algorithms

Data mining algorithms are the collection of techniques in order to perform data mining task. Currently, there are a lot of data mining algorithms for a wide range of data mining tasks. Mainly, these algorithms can be categorized into there groups according to the types of patterns which those algorithms try to discover.

2.2.1 Association Rules Algorithms

Association Rules algorithms (Link Analysis in other words) deal with finding the statistical relations (associations) between two given types of objects that exist in the dataset. In other words, these algorithms find how often events occur together. For example, a statement like "A customer who buys tea from a supermarket will likely buy sugar" is an association. Associations of items in a business unit must be considered carefully in order to develop good strategies.

2.2.2 Classification Algorithms

Classification is assigning the objects in the dataset into a predefined set of classes. Classification is a type of supervised learning, because the set of classes are introduced to the system before executing classification algorithm. Classification of objects in a dataset is very useful both to understand the characteristics of existing objects and to predict the behaviors of new objects. Classification of e-mails, incoming to an e-mail server, into predefined e-mail classes can be an example for this type of data mining. In this way, behaviors of an e-mail server, such as giving priority to e-mails or blocking some ones, can be determined for incoming e-mails.

2.2.3 Clustering Algorithms

Clustering is the grouping of similar objects and a cluster of a set is a partition of its elements that is chosen to minimize some measure of dissimilarity [Kantabutra 1999]. Unlike classification which is a supervised learning technique, clustering is a type of unsupervised learning [Crocker and Keller]. In clustering, objects in the dataset are grouped into clusters, such that groups are very different from each other and the objects in the same group are very similar to each other.

In this case, clusters are not predefined which means that result clusters are not known before the execution of clustering algorithm. These clusters are extracted from the dataset by grouping the objects in it. For some algorithms, number of desired clusters is supplied to the algorithm, whereas some others determine the number of groups themselves for the best clustering result. Clustering of a dataset gives information on both the overall dataset and characteristics of objects in it.

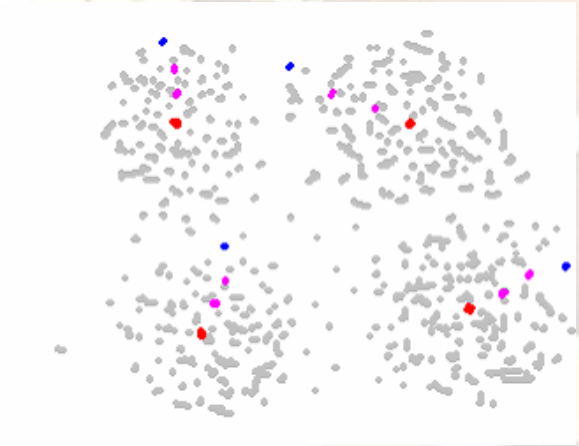
3. SERIAL K-MEANS ALGORITHM

K-means is a data mining algorithm which performs clustering. As mentioned previously, clustering is dividing a dataset into a number of groups such that similar items fall into same groups [Kantabutra 1999]. Clustering uses unsupervised learning technique which means that result clusters are not known before the execution of clustering

algorithm unlike the case in classification. Some clustering algorithms takes the number of desired clusters as input while some others decide the number of result clusters themselves.

K-means algorithm uses an iterative procedure in order to cluster database.It takes the number of desired clusters and the initial means as inputs and produces final means as output. Mentioned initial and final means are the means of clusters. If the algorithm is required to produce K clusters then there will be K initial means and K final means. In completion, K-means algorithm produces K final means which answers why the name of algorithm is K-means.

After termination of K-means clustering, each object in dataset becomes a member of one cluster. This cluster is determined by searching throughout the means in order to find the cluster with nearest mean to the object. Shortest distanced mean is considered to be the mean of cluster to which examined object belongs. K-means algorithm tries to group the items in dataset into desired number of clusters. To perform this task it makes some iteration until it converges. After each iteration, calculated means are updated such that they become closer to final means. And finally, the algorithm converges and stops performing iterations. Expected convergence of K-means algorithm is illustrated in the image below.

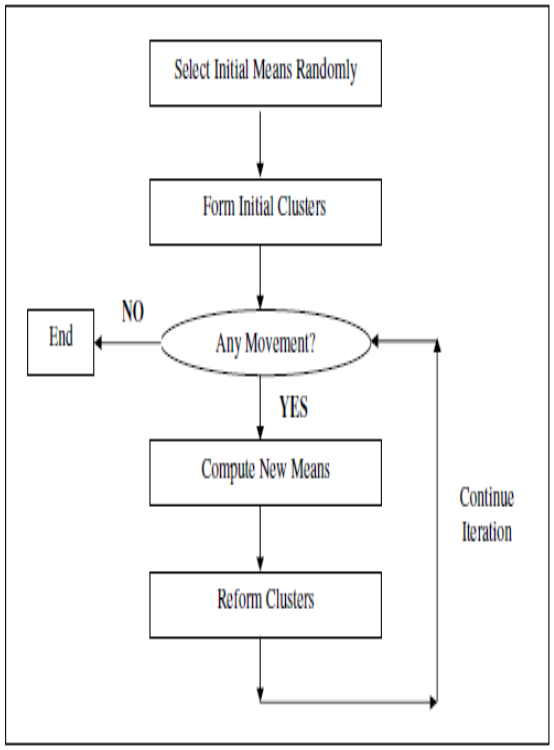


Algorithm converges in three iterations in the illustrated example. Blue points represent the initial means which may be gathered randomly. Purple points stands for the intermediate means. Finally, red points represent the final means which are also the results of K-means clustering. As presented in the illustration, means move to the cluster centroids by each iteration of K-means algorithm. When they reach to the cluster centroids, the algorithm converges.

3.1 Steps of K-means Algorithm

As stated earlier, K-means algorithm takes initial means as input. Each of updates to means in iterations makes those means closer to final means. This is why K-means algorithm converges after a number of iterations. Initial means and produced subsequent means are used to assign objects into clusters. Initially, objects are assigned into clusters that have the nearest mean to them by using initial means which are supplied to the algorithm as input. This is the first iteration of algorithm.

When all objects are assigned into clusters, cluster means are recalculated by using the objects in the clusters. These means are supposed to be closer to final means when compared with initial ones. Next, all objects are reassigned to clusters by using new means. This is the conclusion of second iteration. Probably, some objects will move to different clusters when using new means considering their clusters with the previous means



Steps of K-means Algorithm in Schematic Representation

3.2 Deficiencies of Serial K-means Algorithm

In today's world, company's databases have grown explosively which makes it very time consuming or sometimes impossible to run traditional

data mining algorithms on their data bases. Therefore, serial K-means algorithm will either lack in performance or crash when trying to cluster huge amounts of databases which arises the need of improvement of K-means algorithm in order for the K-means algorithm to cluster large amounts of data in reasonable times.

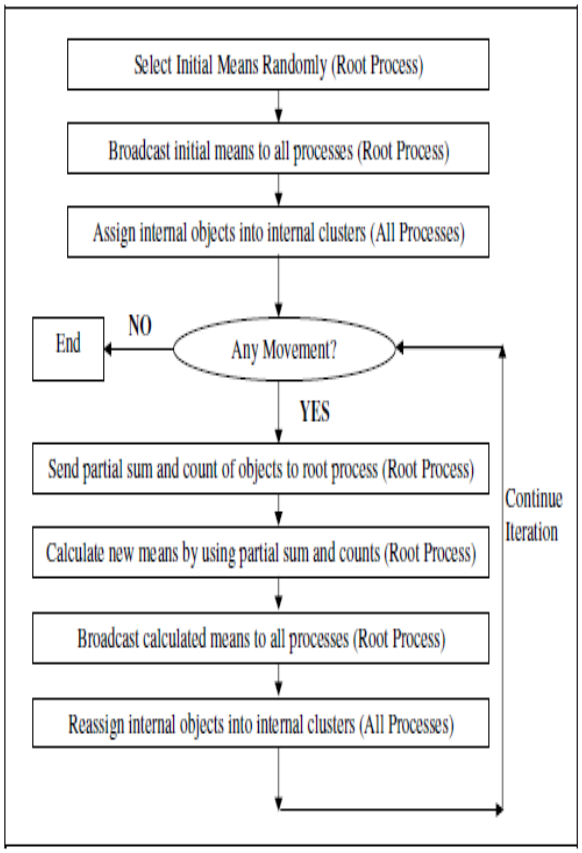
An ideal data mining algorithm should scale well. In other words, the algorithm should produce true results in reasonable time even the database grows to very large amounts. Therefore, some modifications should be done to traditional data mining algorithms in order to make them scale well in case of huge amounts of data.

4. Parallel k-means algorithm

K-means algorithm has been re-designed to run in parallel manner by using the message passing technique of parallelization. parallel K-means algorithm has been developed and implemented in this project by the aim of performance increase when compared with serial K-means. Parallelization of K-means algorithm has been proposed to be a solution for the need of a faster K-means algorithm in order to cluster large amounts of databases in reasonable durations. And also, by using parallel K-means, it has been aimed to gather exactly same clustering results with serial algorithm, since purpose of parallelization in this project is to perform exactly same clustering in shorter duration.

4.1 Steps of Parallel K-means Algorithm

Steps of serial K-means algorithm needs some revision in order to run in parallel manner. In designed parallel version, all the dataset does not remain in one computer's memory; instead, each computer reads and holds an equally divided (by the number of computers used in parallel execution) part of the dataset. This is the point why computers need to communicate for performing the clustering operation. Since each computer has its own memory space and there is no shared memory area, they need to communicate by using message passing. A root computer has been used in parallel algorithm. Root computer is used for the synchronisation of all computers. It broadcasts data to all computers and gathers data from all computers in order to perform K-means clustering. It also performs the same clustering operations as slave computers. It performs synchronization tasks in addition to those clustering operations.



Steps of Parallel K-means Algorithm in Schematic Representation

5. PROPOSED SYSTEM

Due to the large data sets, K-means is not efficient and is not able to handle large data sets very efficiently. As selecting the initial means is very important task in clustering algorithms and it is responsible for the quality of the clusters formed.

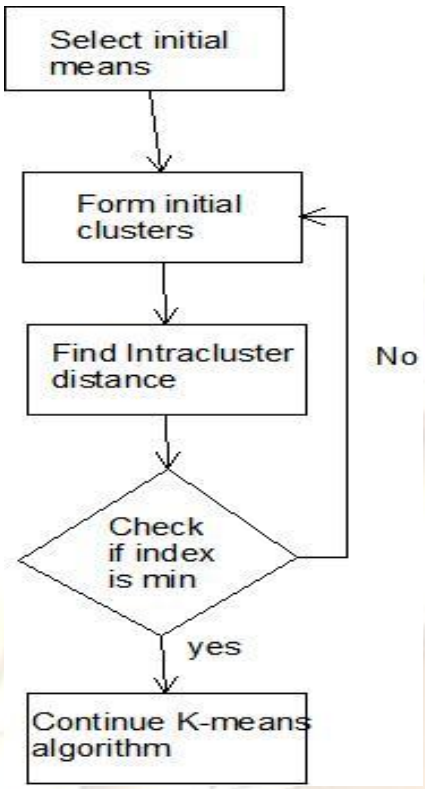
So selecting the initial means and forming the clusters can be done using an index. The distance between intraclusters should be as small as possible. The ratio of minimal intracluster distance to maximal intracluster distance can be taken as $D_{index.}$

$$D_{index} = \frac{Cmin}{Cmax}$$

Where, C_{miin}= minimal intracluster distance.

C_{max} = maximal intracluster distance.

Repeat step recursively until minimum D_{index} is found.



Steps of K-means algorithm are seen below:

- Calculate initial means
- Assign objects into clusters by using initial means
- Calculate D_{index}.
- If D_{index} < Min.Do while objects move to another clusters
  - o Recalculate means of clusters by using objects belonging to them
  - o Assign objects into clusters by using calculated means
- End of while (Convergence of the algorithm)

6. CONCLUSION AND FUTURE WORKS

Main aspect of this project has been to improve the K-means algorithm so that it can perform clustering on large datasets in reasonable durations. When examining serial K-means algorithm, it can be observed that the algorithm deals with all objects in dataset serially which very time consuming especially for large databases. In this project, parallelization of K-means algorithm has been proposed as an improvement for the algorithm.

When examining, it has been observed that selection of initial points also affects the convergence time of Kmeans algorithm. It happens that in one example, execution time of the algorithm has become the half of the previous run time by the change of random initial points. This shows that, a technique for selecting better initial points than random ones may be developed and used in connection with the parallel algorithm as a future work in order to make K-means algorithm even faster.

References

1) B. Thuraisingham, Data Mining: Technologies, Techniques, Tools and Trends,

2) Shared Memory Parallelization of Data Mining Algorithms: Techniques, Programming Interface, and Performance.
[Jin, Ge Yang, and Gagan Agrawal, Member, IEEE Computer Society]

3) Parallel Mining for association rules
[Rakesh Agrawal ,senior member,IEEE and john C. Shafer]

4) Scalable Parallel Data Mining for Association Rules
[Eui-Hong (Sam) Han, George Karypis, Member, IEEE, and Vipin Kumar, Fellow, IEEE]

5) A Compilation Framework for Distributed Memory Parallelization of Data Mining Algorithms
[Xiaogang Li, Ruoming Jin, Gagan Agrawal]