# Enhancing the performance of clustering algorithms using MapReduce computations.

# Abstract

Clustering is an essential data mining technique and tool for analysing big data. However, there are difficulties in applying clustering techniques on big data due to new challenges that are raised with big data. As Big Data refers to terabytes and petabytes of data and clustering algorithms come with high computational costs the question is how to cope with this problem and how to deploy clustering techniques to big data and get results in a reasonable time. However, large scale data clustering has become a challenging task because of the large amount of information that emerges from technological progress in many areas including finance and business informatics. The key requirement to address this challenge are to extract, analyse and process data in a timely manner. Hence parallelization and distribution becomes attractive in terms of speed-up in computation and increased memory available in computation. For such applications, the Hadoop MapReduce framework has attracted a lot of attention. MapReduce is taken as the most efficient model to deal with data intensive problems.

The problem this research paper aims to address is how to use MapReduce framework to implement efficient clustering algorithms for analysing big data. The research mainly contributes to the following aspects. Firstly, an overview of big data is introduced. Secondly, an in-depth study of data clustering is provided, which is the main focus of this research. Thirdly, an overview of the MapReduce framework and how it works to enhance the efficiency of clustering algorithms is also provided.

**Keywords:** Big Data, Data Mining, Clustering, Hadoop, MapReduce

1. **Problem statement and research question**

Clustering can be expensive and time consuming because of its iterative and repeat clustering. Hence parallelizing and distributing becomes attractive in terms of its speed-up in computation and increase memory available in computation (Krishna & Singh, 2014). MapReduce is a programming model which is designed for processing large volumes of data in parallel. Parallel clustering method based on MapReduce is an efficient model to deal with such problems.

The main focus of this research is using MapReduce framework to implement efficient clustering algorithms for analysing big data. Firstly, an overview of big data is introduced. Secondly, an in-depth study of data clustering is provided, which is the main focus of this research. Thirdly, an overview of the MapReduce framework and how it works to enhance the efficiency of clustering algorithms is also provided. In addition, our study also provides an analysis of various clustering algorithms that can be executed in parallel using MapReduce and their challenges.

This work does not aim to introduce new clustering algorithms that are based on MapReduce framework but instead will help establish new knowledge regarding data clustering using MapReduce framework for analysing big data.

**Objectives:**

* 1. To investigate big data.

**1.1.1** To determine what big data is.

**1.1.2** To determine what data mining is and to investigate different data mining techniques.

**1.2** To investigate clustering and explore clustering algorithms and their challenges.

**1.2.1** To determine what clustering is.

**1.2.2** To establish how clusters are computed.

**1.2.3** To investigate challenges with cluster analysis or clustering.

**1.2.4** To determine different categories of clustering algorithms.

**1.2.5** To establish and analyse different types of MapReduce-based clustering algorithms and their challenges.

**1.2.6** To investigate how we can improve the efficiency of existing clustering algorithms.

**1.3** To investigate MapReduce framework and how it works.

**1.3.1** To determine what Hadoop is.

**1.3.2** To determine what Hadoop MapReduce framework is.

**1.3.3** To Establish how MapReduce framework works.

**1.3.4** To argue why MapReduce framework should be used to implement clustering algorithms for analysing big data.

**2. Literature review**

**2.1 Big Data**

Over the last two decades, data in various fields has increased in a very large scale. International Data Corporation(IDC) in 2011, has reported an overall volume of 1.8 ZB(Zettabyte) of both copied and created data in the world. This means that the overall data created and copied increased by nearly 5 times over the years. The recent advances within fields such as Information Technology(IT) has made the generation of more data easier. For example, social networking sites such as Twitter and Facebook have data that span over several terabytes. YouTube has been reported to have a similar size and it produces hundreds of gigabytes per minute.

Every minute, 72 hours on average of videos are uploaded by YouTube users. Facebook generates log data amounting to a sum of 10 petabytes a month. As a result, we are faced with a challenge of collecting and integrating enormous amounts of data from widely distributed sources of data. Other advances that further promote a sharp growth of data include the rapid growth of Internet of Things and cloud computing.

Big data is the term for a collection of data sets so large and complex that it becomes difficult to process using on hand database management tools or traditional data processing applications. Under the explosive increase of data globally, the phrase Big data usually refers to enormous datasets. In comparison with normal or traditional datasets, this dataset is usually unstructured which makes it very difficult and challenging to analyse and interpret the datasets. Therefore, we need to establish techniques to analyse the data and get results within an acceptable amount of time.

Some of the benefits of Big data include the following, firstly it brings about new opportunities for data miners to discover new patterns and values. Secondly, after discovering these new values, Big data helps data miners to gain an in-depth understanding of hidden values within the data. Lastly, it incurs new challenges such as how to manage and organize large datasets. Big data decodes data that has been untouched in order to derive new insight which will be integrated with business operations. However, as data continues to grow exponentially, most of the current data mining techniques become obsolete.

**2.1.1 Characteristics of Big data**

Big data is described by three characteristics which are known as the 3Vs of Big data. These characteristics are volume, velocity and variety. **Volume** refers the large amounts of data that is produced endlessly. Initially, the actual storage of this data was very challenging due to storage costs being high. Over the years, storage costs have decreased and this problem seems to have been kept somewhat at bay for now. Even though this solution is considered temporal, better software solutions needs to developed. Massive amounts of data are generated from platforms such as social networking sites, E-commerce and Smartphones. The data generated can be classified into semi-structured, structured and unstructured data.

**Velocity** refers to the actual speed at which the data is generated or produced. For instance, Google currently processes 1.2 trillion searches per day globally. Facebook produces 2.7 billion like actions/day. **Variety** is concerned with the different formats of the data generated by different platforms. The data generated can be of any time from structured, semi-structured or unstructured. Most unstructured data are generated from satellites and social networking sites. Videos, pictures, documents, spreadsheets and databases are in different formats which cause the data to lose its format.

**2.1.2 Big Data challenges**

The rapid upsurge in data generated in the Big data era brings about serious challenges on the acquisition of data, the storage of data, and the analysis of the data. Outdated data management systems are mostly grounded on relational database management systems(RDBMS). The main problem with such RDBMSs is that, they can only be applied to structured data. Another problem is that RDBMs are increasingly using expensive hardware. Therefore, it is very clear that traditional RDBMSs fail to handle enormous volumes and heterogeneity of Big data.

In the past decade, solutions to these problems have been proposed by researchers from different perspectives. For instance, cloud computing has recently been used to meet the requirements on infrastructure for Big data. Distributed file systems and NoSQL databases have been considered good choices for solutions regarding a more permanent storage and the management of large-scale datasets.

The following section outlines challenges encountered when developing applications for Big data.

**2.1.2.1** **Data representation**

In the era of Big data datasets consist varying levels of heterogeneity in semantics, granularity, accessibility and structure. The main aim of data representation is to ensure that is more meaningful for computer interpretation as well as computer analysis. This implies that an improper representation will diminish the value of the information or data. Furthermore, improper data representation can also hinder effective data analysis. The data structure, type and class and all the integrated technologies which enable efficient operations on datasets is reflected by an effective data representation.

**2.1.2.2** **Redundancy reduction and data compression:**

The sharply increasing data in the big data era bring about high levels of redundancy on the datasets. Data compression and redundancy reduction are two effective approaches used to lessen the unintended cost of the entire system on a premise that the possible values of the data are not affected. For instance, sensor networks generated data is highly redundant. To solve this problem the data can be filtered and compressed.

**2.1.2.3 Data life cycle management**

Since data is generated at unprecedented rates, current storage systems are unable to support or withstand such enormous data. This is due to slow advances in the systems for storage, computing and sensing. Most values hidden in Big data depend on the freshness of the information. Therefore, there is a need for an analytical system which will determine which data should be stored and which data should be discarded.

**2.1.2.4 Analytical mechanism**

The Big data analytical system should be designed to process enormous amounts of heterogenous information or data in a reasonable amount of time or within a time period which is acceptable. However, current RDBMSs cannot be expanded and are not scalable and therefore are unable to meet the performance requirements. Non-relational databases have certain advantages as opposed to typical RDBMSs in processing unstructured data and have dominated the mainstream of data analysis. Even though non-relational databases have shown improvements in performance as opposed to traditional RDBMSs, non-relational databases still have problems with regard to their performance therefore more research is still needed on the in-memory databases which will have a significant improvement in performance.

**2.1.2.5 Data confidentiality**

Service providers of Big data are unable to effectively maintain and analyse these huge amounts of data due to their limited capacity. They depend on professionals and tools to analyse the data which results in an increase in safety risks. For example, information about transactions usually consists of a set of complete operating data which can be used to drive key business processes. The information may be made up of sensitive information such as credit card numbers. Therefore, it has been deemed necessary to employ a third party who will analyse Big data of an organization when proper precautions have been taken to protect such sensitive information.

**2.1.2.6 Energy Management**

In order to process this enormous amount of data or information, mainframe computers have been used. The problem with mainframe computers is that they consume a lot of energy and therefore have drawn .so much attention from both environmental and economical perspective. As a result of the rapid increase in volumes of data generated from different sources, the processing, storage and transmission of data will consume a lot of electric energy. To lessen this, power consumption control mechanisms at a system level has to be in place for Big data.

* + - 1. **Expandability and Scalability**

Analytical systems must be designed in such a way that they support present as well as future datasets. The algorithms for Big data analysis must be designed to process present as well as future expanding complex datasets.

* + - 1. **Cooperation**

Big data analysis is achieved by combining experts from various fields who cooperate to find useful patterns on the information. A Big data network architecture must be put in place to help scientists and engineers in different fields to access different kinds of data.

* + 1. **Mining Big Data**

Data mining refers to a process of extracting hidden predictive information from data, which helps companies to focus on important information in their data warehouse. Data mining tools are used for predicting future behaviours and trends to enable businesses or organizations to make knowledge driven and practical decisions. The contribution of data mining is manly done through a database search to evaluate hidden patterns, predictive information which may be missed by experts, due to the information being outside their expectations. The rapid upsurge in data generated in the Big data era will mean that extracting the required patterns hidden on this data will become very tedious if conventional approaches are used. In addition to that, conventional methods are time consuming and less efficient hence parallel processing of data has gained major importance.

Some data mining techniques include Anomaly detection, Association rule learning, Clustering, Regression, Classification and Summarization. This paper is mainly concerned with clustering which is a process of organizing objects in groups whose members contain some kind of similarity. Traditional clustering techniques cannot cope with this huge amount of data because of its complexity. Therefore, the main goal is to scale up and speed up clustering algorithms with minimum sacrifices to the clustering quality.

**The following section outlines different types of data mining techniques.**

* + - 1. **Classification**

Classification which employs a set of pre-classified examples to develop a model that can classify the population of records at large. Credit risk applications and fraud detection are particularly well suited to this type of analysis. This technique frequently uses decision tree or neural-based classification algorithms. The data classification stage involves leaning and classification.

**2.1.3.1.1 Types of classification methods:**

2.1.3.1.1.1Classification by decision tree induction

2.1.3.1.1.2Bayesian Classification

2.1.3.1.1.3Neural Networks

2.1.3.1.1.4 Support Vector Machines (SVM)

* + - 1. **Clustering**

Clustering is a technique of identifying similar classes of objects. The use of clustering algorithms allows for the identification of dense and space regions in the object space. Therefore, the distribution pattern and the correlation between data attributes can be determined. The classification technique can also be used like clustering for distinguishing groups but it becomes very costly hence clustering is used as a pre-processing technique for attribute subset selection and classification.

**2.1.3.2.1. Types of clustering methods**

2.1.3.2.1.1 Partitioning methods

2.1.3.2.1.2 Density based methods

2.1.3.2.1.3 Hierarchical methods

2.1.3.2.1.4 Grid-Based methods

2.1.3.2.1.5 Model based algorithms

* + - 1. **Prediction**

Regression analysis models the relationship between one or more independent variables and dependent variables. Independent variables in data mining are all attributes which are already known and dependent variables are attributes to be predicted.

**2.1.3.3.1 Types of regression methods**

**2.1.3.3.1.1** Linear Regression

**2.1.3.3.1.2** Multivariate Linear Regression

**2.1.3.3.1.3** Nonlinear Regression

**2.1.3.3.1.4** Multivariate Nonlinear Regression

* + - 1. **Association rule**

Association and correlation are used to obtain frequent item set findings among enormous data sets. This technique helps decision makers in businesses to make wise business decisions. Association rule algorithms should be able to generate rules that have a confidence level that is less that one. However, the number of possible association rules for a certain dataset tend to be very large. In addition to that, a large amount of the generated rules has little or no value at all.

**2.1.3.4.1 Types of association rule**

2.1.3.4.1.1Multilevel association rule

2.1.3.4.1.2Multidimensional association rule

2.1.3.4.1.3Quantitative association rule

* + - 1. **Neural networks**

Neural network refers to a set of connected input/output units and each connection has a weight to represent it. Neural networks have a strong ability to derive meaning from complicated or imprecise data. It is also used to detect trends and for the extraction of patterns that are very complex to be noticed by humans. Neural networks are also well suited for prediction or forecasting needs.

* + - * 1. **Types of neural networks**

Back Propagation

* + - 1. **K-Nearest Neighbours**

The K-Nearest Neighbours technique is a simple machine learning algorithm used for predicting a certain response value for a give observation using responses of all the observations in that observation’s local neighbourhood. This technique can also be used categorical response for classification and with a continuous response for prediction.

* + - 1. **Regression**

Regression is a data mining technique that predicts a certain number. Income, distance, weight and temperature can all be predicted using regression techniques. For instance, given the age, weight and other attributes of a child, a regression model can be used to predict the child’s height. The dataset might contain attributes such as age, weight, family history and other attributes. The height of a child would be the target, and the other attributes would be predictors.

During the training phase, a regression algorithm is used to approximate the actual value of the target as a function of all the predictors in the dataset. All the relationships between the predictors and target are abridged in a model. This model can then be applied to another dataset with target values unknown.

**Types of regression methods**

Multivariate linear regression

Nonlinear regression

Multivariate nonlinear regression

**2.1.3.8 Summarization**

It provides a more compact representation of the data set, including visualization and report generation. Data summarization provides a capacity to give data consumers a generalized view of enormous amounts of data.

**2.1.3.9 Anomaly detection**

This is the identification of the unusual records or data errors. In other words, anomaly detection identifies unusual patterns that do not conform to the expected behaviour, which are called outliers. One of the applications of anomaly detection in business include intrusion detection, fraud detection and fault detection.

**2.2 Cluster Analysis**

Cluster analysis is one of the techniques used for mining big data to extract useful patterns. The goal of data clustering is to determine the intrinsic grouping in a set of unlabelled data. Traditional clustering techniques cannot cope with this huge amount of data because of their complexity. However, largescale data clustering has become a challenging task because of the large amount of information that emerges from technological progress in many areas, including finance and business informatic. Accordingly, researchers have dealt with parallel clustering algorithms using parallel programming models to address this issue. MapReduce is one of the most famous frameworks, and it has attracted great attention because of its flexibility, ease of programming, and fault tolerance.

2.2.1 **Categories of cluster analysis**

**Hierarchical methods**

This approach involves creating various partitions and then evaluate the using a specific criterion. For instance, suppose there is a dataset of ‘n’ objects and the partitioning methods creates ‘k’ partitions of the given dataset. Each partition represents a cluster with k≤n. Hence, data will be classified into k number of groups which should satisfy the following requirements.

1. Each partition should consist of at least one object
2. Each object is allowed to belong to a single group

**Portioning methods**

The technique involves creating hierarchical decomposition of a set of object or data. Hierarchical methods are classified based on how the hierarchical composition is formed. This can be achieved in two approaches

1. Divisive approach

This technique is also called the top-down approach. This technique starts with all the data objects in a single cluster. The cluster is then split up into smaller clusters in a continuous iteration. This is done until only one object is left on a cluster or the termination condition holds

1. Agglomerative approach

This technique is also called bottom-up approach. This approach starts with one data object making up a single group. Group that are close to each other are then merged in a continuous iteration. This will be done until all the groups are merged or the termination condition holds.

**Density-based algorithms**

This approach is based on the density and complexity functions. This technique works by growing a given cluster as long as the neighbourhood’s density exceeds some given threshold.

**Grid-based methods**

This technique is based a multiple-level granularity structure. This means that all objects that are together will form a grid. The object space is quantized into a finite number of cells which form a grid structure. This approach has an advantage of having faster processing time.

**Model-based Methods**

This approach forms clusters by clustering the density function. For each of the clusters, a model is hypothesized and the idea I to find the best fit for that specific model. With this approach, the number of clusters can be deduced automatically based on standard statistics, taking noise and outliers into consideration. Hence, this approach yields very robust clustering methods.

**Constraint-based method**

This approach works by incorporation of application-oriented or user constraints. Constraint in this case refers to the properties of the user’s desired clustering or user expectation. These constraints can be specified by the application requirement or the user. Constraints are also useful for providing us with an interactive way of communicating with the entire clustering process.

2.2.2 **Clustering algorithms applicable for MapReduce**

**K-Means algorithm**

K-Means is considered on the most widely used clustering algorithms in clustering. This algorithm takes an input n, which refers to the number clusters that the algorithm should create and k which a set of objects. The first step of the algorithm involves randomly choosing the cluster centres and then centroids are calculated using distance functions like Manhattan distance or Euclidian distance. These steps will be performed iteratively until clusters are created.

K-Means can be implemented using MapReduce Framework as follows.

Map function

Input data is stored by the HDFS as a sequence of <key, value> pairs with each <key, value> pair representing a record. Data is distributed across all the mappers after splitting. Cluster centre information is stored on a global variant called “centres”. The closest centre for each record is then computed by the mapper.

Combine function

The combiner is responsible for combining intermediate data of the same mapper. This intermediate data will be located in a local disk of the host machine. Value points are also summed up in this phase.

Reduce function

All the output of the combine function is given to this function as input. At this stage, a summation all the values from all the nodes is produced and the results will be obtained. Thereafter, new centres are generated for further iterations.

NBBBBBBBBBBB PICTURE OF MAPREDUCE BASED K-MEANS

**DBSCAN (Density Based Spatial Clustering of Applications with Noise) Algorithm**

This is a density-based clustering algorithm. The basic idea behind this algorithm is that points that form a dense region, are grouped together to form a cluster. In order to find this dense region, a fixed threshold value is used. DBSCANis used mostly in research. The main advantage of this algorithm lies in the ability to handle noise data efficiently. However, even though this algorithm is more efficient than most clustering algorithms, the algorithm cannot handle large amounts of data. In addition to that, it suffers from bottleneck which is the main reason MapReduce framework is applied to this algorithm.

The MapReduce approach of DBSCANworks as follows. Firstly, the dataset has to be partitioned and distributed among the nodes for processing. A global index structure is required by DBSCANbut it creates an extra communication cost. Therefore, a distributed index is used for efficient parallel processing. Data points belonging to the same cluster are allowed to be located on different nodes but these scattered points have to be merged. During the first phase which is called “Partition with reduced boundary points”. In this phase, boundary points are minimized and the input data is divided among the different nodes on the cluster. The main benefit of minimizing boundary points, is that it increases the efficiency of clustering and it also improves merging. The second phase of this MapReduce based approach is called the DBSCAN-Map. On this phase, the DBSCAN-Map algorithm is executed on every node on the cluster. KD-tree space index is used on this execution. The algorithm is executed on assigned and divided data locally. The third phase is referred to as “DBSCAN-Reduce”. Point indexes between partitions are found on this phase. In addition to that, the clustering ID of the points is calculated on this phase. The fourth phase is known as “Merge Result”. This phase involves getting results from the DBSCAN-Reduce phase and merging them. Based on boundary points obtained after merging, the global structure of clusters is established. The final phase of the algorithm is known as “Relabel”. On this phase, the results from local clustering on each data partition are relabelled and then global clusters are identified.

**Canopy clustering**

This algorithm falls under the category of unsupervised clustering algorithms. The algorithm is mainly used as a pre-clustering method which means the output produced by this algorithm is given as input to another algorithm. Pre-clustering improves the efficiency of major clustering algorithms when processing enormous datasets. The algorithm has six steps which are outlined below.

1. List containing data points which are named X is formed.
2. Two threshold values T1 and T2 are selected where T1>T2
3. A data point is selected randomly, representing a canopy centroid
4. Distance d from the centroid is calculated for all points
5. If distance d is greater than threshold value T1(d>T1), that point should be added to the canopy. Otherwise, if distance d is less than threshold value T2(d<T2), remove the point from the list X.
6. Repeat the steps III to V until all the points have been visited.

**MinHash clustering**

MinHash clustering belongs to the category of Locality Sensitive Hashing(LSH) algorithms. It is more suited in situations where clustering should be done based on various dimensions of data points. This approach is a type probabilistic method. The overlap between a set of items is directly proportional to the probability of a data point being assigned to a single cluster.

The number of clusters

If the number of class labels is not known beforehand, the identification of clusters becomes a difficult task. The number of clusters have to be analysed carefully to produce the correct results otherwise, similar tuples may be broken into many or heterogonous tuples may merge which could be catastrophic if the approach used is hierarchical. This is because in a hierarchical approach, if tuples get wrongly merged, that action cannot be undone.

Identification of a distance measure

Identification of distance measure for categorical attributes is more difficult than identifying a distance measure for numeric attributes. Numerical attributes use standard equations like Manhattan, Maximum distance measure and Euclidian as distance measures. All the standard equations are a special case of MinKowski distance.

Lack of class labels

With real datasets, data should be distributed in such a way that class labels can be located easily.

Structure of the database

Real life data does not always contain clusters that are identifiable. In addition to that, the results produced by an algorithm may be affected by the order in which the tuples are arranged if the distance measure used is not perfect. With data which does not have a structure, identification of an appropriate number of clusters will not produce good results. For instance, missing values can exist for tuples, variables and also randomly in tuples and attributes.

Types of attributes in the database

Databases may not contain distinctively categorical or numerical attributes. They can also contain other types like binary, ordinal, nominal etc. In order to make calculations simpler, these attributes have to converted to categorical.

Choosing initial clusters

When utilizing the partitional approach, most algorithms require k initial clusters to be randomly chosen. If the initial clusters are not chosen properly, after a few iterations of the algorithm, clusters may be left empty.

* 1. **Cluster analysis for big data**
  2. **Apache Hadoop and MapReduce**

Apache Hadoop is an open source software framework which uses the MapReduce software framework for distributed processing and storage of big data. The framework is made up of computer clusters which are built from commodity hardware. The core of Apache Hadoop consists of two parts, namely, the processing part and the storage part. The processing part is called the MapReduce model while the storage part is known as the Hadoop Distributed File System (HDFS). Hadoop works by splitting files into big blocks which are then distributed across different nodes in a cluster. Packaged code is then transferred into nodes to process the data in parallel. This approach uses the principle of data locality to process to process the dataset more faster and efficiently. The principle of data locality allows nodes in a cluster to only manipulate data they have access to.

* 1. **MapReduce programming**

MapReduce processes data points over a distributed environment which consists of commodity machines. This occurs in two phases, namely, the Mapping phase and Reduce phase. Mapping is defined by the Mapping function. The input data are split into map functions and then computations are performed which results intermediate results being produced. The intermediate results are in form of key/value pairs.

After the intermediate results are generated, the data is then shuffled into corresponding Reducer to perform the reduce tasks. The Reduce phase is defined by the Reduce function which takes in a single key and processes the specified function on its associated values at a time. Overall, the data is processed through the following steps as outlined in figure 1.

**Map function** – Responsible for reading data points as input. It then generates intermediate key/value pairs. The key represents the group number of values and the value is associated with a key.

**Combiner function** – Intermediate results that have significant repetition in their own Map task are passed through the combiner function. Therefore, this function is responsible for partial reduction before intermediate key/value pairs are passed to the reducer.

**Partition function** – Responsible for performing partitioning.

**Reduce function** – The output produced by the Map tasks is the input of the Reduce phase. For each key, the Reduce task is applied once, and its values are then processed with respect to the user-defined Reduce function which is called by each reducer once for each key.