Performance of K-means in Hadoop Using MapReduce Programming Model

1st Engelbertus Vione[1], 2nd J.B. Budi Darmawan[2]
{raikeiji@gmail.com[1],b.darmawan@usd.ac.id[2]}

Department of Informatics, Faculty of Science and Technology, Sanata Dharma University, Mrican, Tromol Pos 29, Yogyakarta 55002, Indonesia[12]

Abstract. Hadoop which is one of the big data framework uses MapReduce programming model to analyze data. Mahout is a data analysis library that has the ability to use MapReduce programming. One of the clustering algorithms supported by Mahout is K-mean. The researchers are interested in observing the performance speed of applying the K-mean algorithm from Mahout to cluster liver disorder data set from UCI with changes in the configuration of the number of slave nodes using Hadoop. This study uses 4 computers with a configuration of 1 master node and 3 slave nodes in the Hadoop cluster that runs on the local network. The results of the average speed of the K-Means process using 344 data sets indicate that increasing the number of slave nodes from one to three will increasenon-linearly the speed of the computational process.

Keywords: big data, Hadoop, MapReduce, Mahout, K-means.

1 Introduction

Hadoop is a big data framework that can store data on a large scale without regard to the structure of the data. Large data collections can be processed and analyzed to get values from data. The results of data analysis are in the form of information that can be used to support a decision making on the organization. Hadoop uses the MapReduce programming concept to process data into information. MapReduce is capable of computing in parallel and distributed on the Hadoop system.

Mahout is a library that uses MapReduce programming concept and can adapt to the Hadoop system. Therefore, Mahout can be used to analyze very large data. Mahout provides data mining algorithms to analyze data and K-Means is one of the algorithms provided by Mahout. K-Means analyzes data by grouping data based on similar properties.

The accuracy of K-mean clustering using Mahout has been observed to cluster massive dataset [1]. However, the speed performance of the computation process of K-mean clustering using Mahout has not been observed. In this paper, researches are interested in observing the performance speed of applying the K-mean algorithm from Mahout with scenarios in the configuration of the number of slave nodes using Hadoop.

ICSTI 2018, October 19-20, Malaysia

DOI 10.4108/eai.19-10-2018.2282545

2 K-means

K-mean is one of well-know clustering algorithms. The k-means algorithm takes the input parameter k and partitions a set of n objects into k clusters. The characteristic of resulting clusters have high intracluster similarity but low intercluster similarity. It proceeds as follow. First, it choose k of objects randomly as centroids, center of each cluster. For each of the others object, an object is assigned to the cluster to which it is most similar, based on the distance between the object and the centroid. It then compute the new mean for each cluster to become new centroid. This process iterates until the criterion function converges or it reach maximum iteration [2].

3 Hadoop

Apache Hadoop is a framework which allows distributed processing from large data using cluster computers with a simple programming model [3].In the Hadoop cluster, data is distributed to all nodes of the cluster. Hadoop Distributed File System (HDFS) will split large data files into chunks that are managed by different nodes in the cluster. In addition each chunk will be replicated to several machines, so failure of one machine does not cause data to become unavailable. Although chunk is replicated and distributed to multiple machines, these chunks have a single namespace name.

Data in Hadoop programming framework is conceptually record-oriented. Input file will be broken down in rows or other formats specific to the application. Every process that runs on a node in the cluster will process a subset of this record. Then Hadoop Framework schedule processes on site where record is located using knowledge from distributed file system. Because files are scattered in a file system distributed as chunk, every calculation process that runs on a node operates on a subset of data. Which data is operated by a node is selected based on the locality of the node, most data read from the local disk to reduce tension network bandwidth and prevent network transfers unnecessary. This allows Hadoop achieve high data locality that can produce high performance [4].

Hadoop reduces the amount of communication done by the process, this is because every record individually processed by an isolated task one with others. Programs must be written in certain programming models, namely MapReduce. In MapReduce, records are processed in isolation by a task called Mapper. Output from Mapper brought together to a second set of tasks called the Reducer, where the result of the Mapper can be combined differently like that shown in Figure 1.

Separate nodes in a cluster Hadoop still communicates with each other. Fractions of data can be tagged with the name of the key which is inform Hadoop how to send this data to the next destination node. Hadoop internally manage all data transfers and problems cluster topology.

Conceptually, the MapReduce program transform the list of input data elements into list of output data elements. A MapReduce program has two list processing, namely map and reduce. The first phase of the MapReduce program called mapping. A list of data elements is given in the Mapper function that will transform each individual element to the individual output data element. List of input strings here it is not modified but produces a new strings that are part of a list new output.

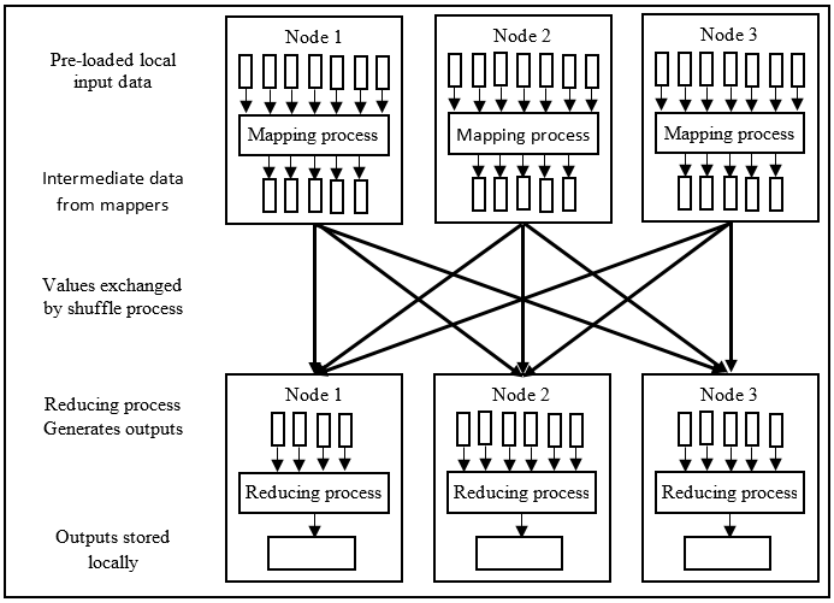

Fig. 1. Mapping and reducing tasks run on nodes [4].

Reducing allows collection of values together based on the same key value. The Reducer function accepts an iterator from the input value of an input list. Reducer then combine these values together, generate a single output value like presented on Figure 1. Reducing is often used to produce data summary or aggregate large volume of data.In MapReduce, all output values are not usually reduced together. All values with the same keys is presented as a single reducer.

4 Manhout

Mahout, an Apache machine learning library, has a computational basis for recommender engine, clustering, and classification. In addition, Apache Mahout is scalable. It can be used as a choice of machine learning tools when the data collection that will be processed is very large so it cannot be stored on a computer. Mahout is written using the Java language and some of the Mahout are developed on the Apache Hadoop Distributed computing project. Mahout is a framework that is suitable for use and adapted by developers. Mahout places scalability at the highest priority. The latest machine learning method is applied to the level of scalability. The open source Mahout library is used in the Hadoop environment, and Mahout supports the MapReduce computing concept [5].

4.1 K-Mean MapReduce in Mahout

The K-Means clustering process in the Mahout library that uses the MapReduce programming concept can be divided into the following phases [1]:

InitialThe input data can be split into several data collections. A list of data sub-dataset are form into the <Key, Value> list. The list of data collections will be entered into the map function. The next process is selecting k point randomly from dataset as initials clustering centroids.

MapperUpdate the cluster centroid. In the Mapper phase, the process is continued by calculating the distance between each data item with the K centroid. Arrange each data to the nearest cluster until all data have been processed. The output of the calculation is the data item with the format <ai, zj>. Ai is the center of the cluster zj.

Reducer The process continues at the Reducer phase. The first process is to read the output of data items <ai, zj> from the Mapper phase. Collect all the data record to produce k clusters with the data point. The process then calculates the average value of each clusters. The output of the process will be used as the value of the new centroid. Next, the system will calculate the value of the new centroid with the original centroid in the same cluster. If the centroid value is smaller than the threshold or the number of iterations of the algorithm has reach the maximum, the algorithm will stop. Otherwise, the new cluster centroid will be used to update the original centroids. Return to map stage, and continue the algorithm until merging.

The K-Means process in the Mahout library using the MapReduce programming model can be visualized in Figure 2.

5 Methode

Dataset used in this paper from UCI machine learning about liver disorder data set. The dataset consist six attribut. And the seventh atribut as the label. We use 344 data item for both label.

Using K-Mean in Mahout library, we use some configuration as follow: Euclidian distance measure is used, the maximum of iteration is 100, the number of cluster (k) is 2 to adapt the number of label in the dataset, and the execution method will use mapreduce.

Performance analysis are evaluated by running K-Means using the Mahout library on Hadoop cluster consist of one master node and three slave node using personal computer with i3 processor connected on local network. We consider scenarios of 1, 2, and 3 slave node for our computations.

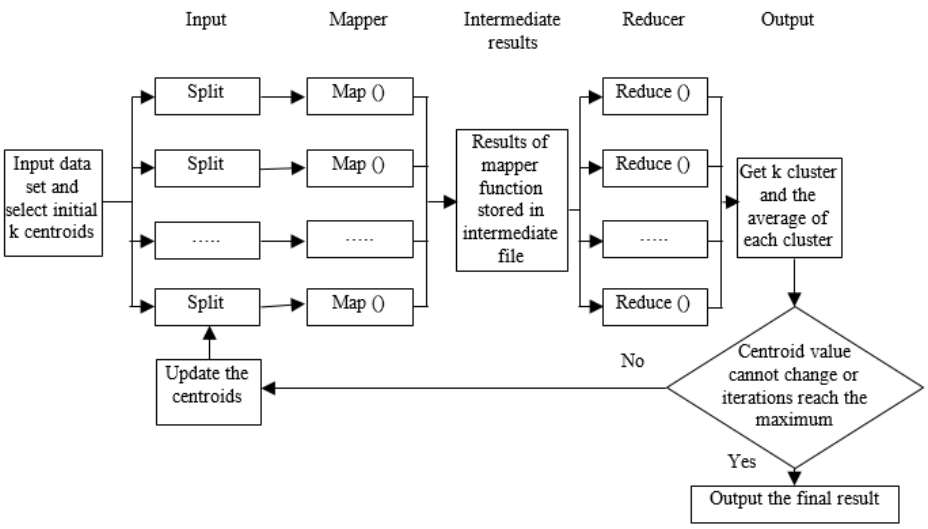

Fig. 2. MapReduce Process using K-means Algorithm [1].

6 Computational results

The performance measurement is conducted by running K-Means computation using the Mahout library in Hadoop 10 times for each different number of slave nodes. Average values are used to evaluate performance results. For a K-Means clustering using 344 liver disorder data we record the total time for each computation in Table 1. The total time for all computations are viewed in Figure 3. Figure 3 indicates that increasing the number of slave nodes from one to three will increase the speed of the computational non-linearly.For 344 record data set, the speed of computation using from 1 to 2 slave node increases significantly. On the other hand, the speed of computation using from 2 to 3 slave node increases less significantly. The size of data set which is relative small may impact this speed performance increment pattern. The increment pattern with larger dataset should be investigated in the future work.

Table 1.Total time for K-Means computation in Hadoop for each different number of slave nodes.

Number of slave node	Total time(second)
1	295.67
2	279.59
3	277.97

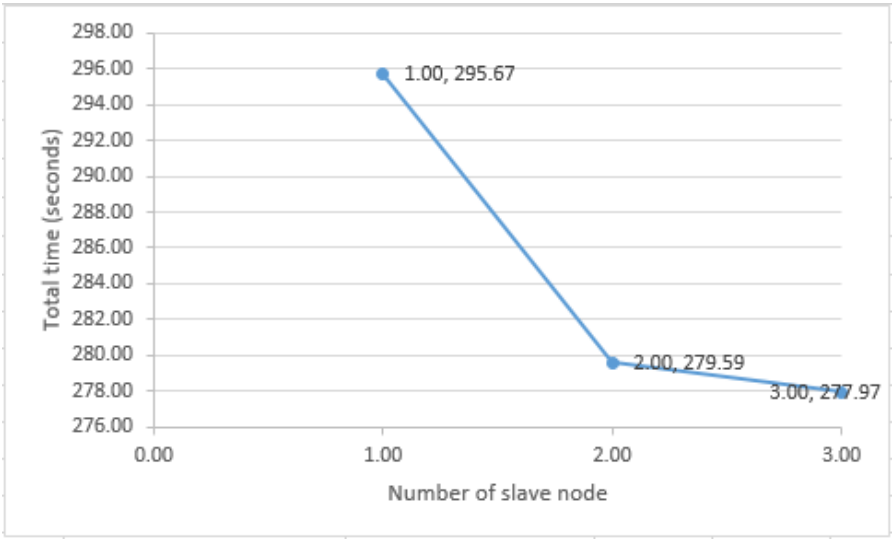

Fig. 3. Total time elapsed in a K-Mean computation for several slave node settings.

The size of data set which is relative small may impact this speed performance increment pattern. The increment pattern with larger dataset should be investigated in the future work.

Conclusion

We have computed several scenarios of number of slave node in Hadoop for K-Mean clustering using Mahout library with 344 record liver disorder dataset.We have obtained a strategy for K-Mean computations using Hadoop consisting of PCs with i3 processors. The increasing number of slave nodes from one to three will increase the speed of the computation non-linearly. For future direction, we will investigate some computation for K-Mean clustering with Mahout library using larger dataset.

References

[1] N. Vishnupriya and S. Francis, .: "Data Clustering using MapReduce for Multidimensional Datasets. International Advanced Research Journal in Science, Engineering and Technology," pp. 39–42, 2015.
[2] J. et al Han, Data Mining Concepts and Techniques Third Edition. Morgan Kaufmann, 2012.
[3] Apache, "No Title," 2018.
[4] Yahoo, "No Title," 2018.
[5] S. Owen, R. Anil, T. Dunning, and E. Friedman, Mahout in Action. 2012.