An Improved Parallel K-Means Algorithm Based on Cloud Computing

Dongbo Zhang[1(✉)] and Yanfang Shou[2]

[1] Department of Computer Science, Guangdong University of Science and Technology, Dongguan, China
neversurrender1314@163.com
[2] Guangzhou Institute of Modern Industrial Technology, South China University of Technology, Guangzhou, China

Abstract. In this paper we presented CK-means clustering algorithm based on improved K-means algorithm and the Canopy algorithm, which uses MapReduce programming model of Hadoop platform. The experimental results prove that the CK-means algorithm has a good advantage in the processing of large data sets, in the acceleration ratio, accuracy, expansion rate, and the effect of the algorithm after deploying on the Hadoop clusters.

Keywords: Cloud computing · MapReduce model · K-means clusters · Canopy algorithm

1 Introduction

Birds of a feather gather together, Clustering algorithm [1] is to research how to become a collection of physical or abstract objects to multiple classes or groups which Composed of similar objects. The objects in the same cluster are as similar as possible, while the objects in different clusters are as different [2] as possible. With the development of social information, the data on the network is growing exponentially, the global daily data generated by 2 EB [3]. How to retrieve valuable information from mass data has become our most urgent target. Clustering analysis, as an important part of data mining, has been widely used in various fields and has achieved some results, however, with the rapid growth of data scale, clustering algorithm has been transferred from serial to parallel, from single machine to clusters. At present, there are some literature about on the parallel clustering algorithms. The paper [4–6] proposed the idea of K-means clustering algorithm based on MapReduce. The paper [7] proposed a parallel clustering algorithm PK-means based on MapReduce, where the Map function computes the distance between the data object to the center point, and then re marks the new clustering category. The reduce function calculating the new clustering center based on the intermediate results. The paper [8–10] implemented the K-means clustering algorithm based on MapReduce.

At present, the classical K-means clustering algorithm [11, 12] has the advantages of simple structure, flexible change, easy hardware implementation, but the K-means algorithm still has the following disadvantages:

K. Li et al. (Eds.): ISICA 2015, CCIS 575, pp. 312–320, 2016.
DOI: 10.1007/978-981-10-0356-1_32

(1) In the process of clustering, the local traps are prone to occur.
(2) In the clustering calculation, the number of iterations is increasing rapidly, and the time consuming is increasing. For better to solve these problems, this paper proposes a new algorithm of CK-means and Canopy based on the in-depth study of Hadoop technology.

2 Related Works

2.1 K-Means Algorithm

K-means is a classical clustering algorithm, Its main idea is we select k data points as the center of the initial K cluster randomly from the original target data sets, then calculates the distance between the other non central data points to the K cluster center. According to the distance from the center, select the nearest cluster, and then assign the data to the cluster, and then the data points are assigned to the cluster, and the process is repeated.

2.2 Canopy Algorithm

Canopy algorithm: For mass data, the data points are divided into some overlapping clusters by using the distance measurement method. And then, clustering data by using the method of calculating the accuracy of the points in Canopy.

3 CK-Means Clustering Parallel Algorithm

CK-means is a algorithm based on MapReduce programming model, which can be divided into two sub tasks. And then implemented based on MapReduce model in the natural order. The inputs of each subtask is the outputs of the previous subtask. The first subtask is to compute the similarity of the data set object based on the MapReduce programming model by using the Canopy idea. And then, put the data points with the similarity into a subset, which is Canopy. However, we no longer carry calculation of the similarity between objects of Canopy, which improved the efficiency of the clustering. Finally, we assign the number of Canopy in this subtask as the initial K value of the next subtask to avoid the blindness of the initial value setting. The second subtask is to make clustering analysis which are based on MapReduce programming model in the "Canopy" by Using the "internal error square and" (see Definition 3) "extreme point" principle (see Definition 2), and combined binary chop. The CK-means executing flow as Fig. 1(a) and (b).

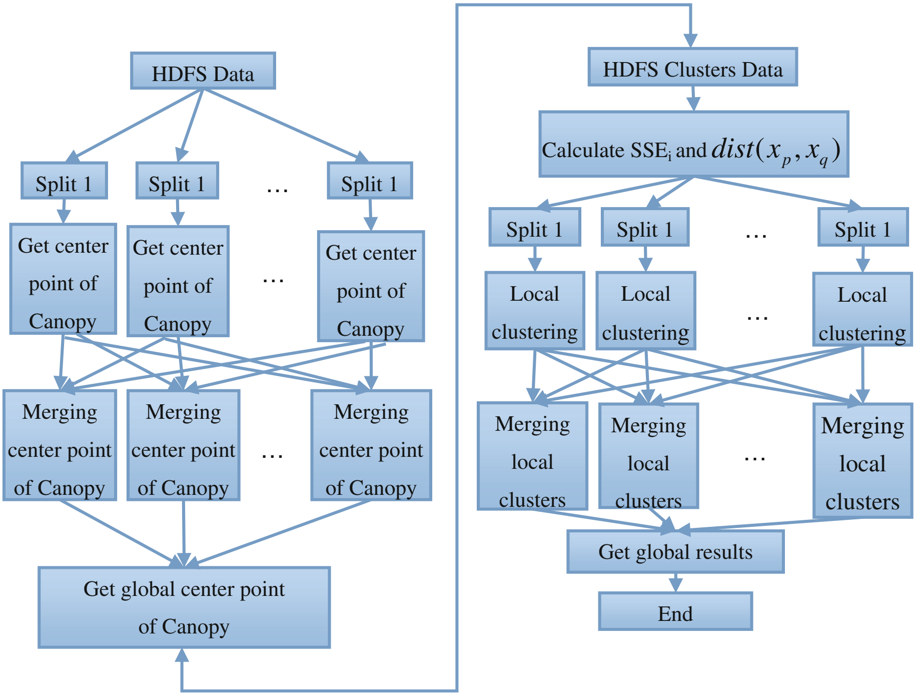
HDFS Data
Split 1
Split 1
…
Split 1
Get center point of Canopy
Get center point of Canopy
…
Get center point of Canopy
Merging center point of Canopy
Merging center point of Canopy
…
Merging center point of Canopy
Get global center point of Canopy
HDFS Clusters Data
Calculate SSE$_i$ and $dist(x_p, x_q)$
Split 1
Split 1
…
Split 1
Local clustering
Local clustering
…
Local clustering
Merging local clusters
Merging local clusters
…
Merging local clusters
Get global results
End

Fig. 1. (a). CK-means executing flow 1. (b). CK-means executing flow 2

3.1 Definitions and Concepts

Definition 1 (Canopy definition): Given data set U $= \{\mathrm{u_i}|\mathrm{i} = 1, 2, \ldots, \mathrm{n}\}$, as to $\forall x_i \in U$ meet $\{c_j | \exists ||x_i - c_j||\} \leq T_1$, $c_j \subseteq U, i \neq j\}$, The x_i collection called Canopy collection, c_j is the center point of Canopy, T_1 is the radius of Canopy collection.

Definition 2 (SSE_i): Sum of the squared errors of cluster C_i, also is the square of the distance between all points in cluster C_i and the center of the cluster. The calculation formula is as $SSE_i = \sum_{x \in c_i} dist(c_i, x)^2$.

Among them, c_i is the center of the cluster C_i.

Definition 3 (Limit point principle): Given cluster $C = \{u_i | i = 1, 2, \ldots, n\}$, $\exists x_p, x_q \in C$ make $dist(x_p - x_q) = \max\{dist(x_i - x_j) | \forall x_i, x_j \in C\}$,

Then x_p, x_q is the limit point of cluster C. And consider x_p, x_q as the initial center of cluster C is limit point principle. Among, $dist(x_p - x_q)$ is limit distance.

3.2 Canopy Algorithm Based on MapReduce

The Canopy algorithm starts with a collection that contains some data points and an empty Canopy list in the Map stage. Then, it iterating for classification according to the distance threshold, and generate Canopy collection in the iterative process. So we got the Canopy collection. In the Reduce stage, it merging the collection W_i with all node, and get the union set w, then making Global Canopy clustering in set w. Repeating the above process until the data set is empty. The algorithm output the number of clusters K finally. This value of K will be the input value for the next subtask.

In the Map phase, The data set of each node in the cluster is needed to clustering, and output Canopy collection W finally. The collection W in Canopy (V, N) will be used as the input to participate in the Reduce.

3.3 K-Means Algorithm Based on MapReduce

Algorithm process: First, the algorithm consider all data sets as cluster V and put it into cluster S in order to get the cluster k. Then the algorithm take out a cluster which must meet the limit point principle from cluster S by K-means clustering algorithm. It making two points clustering by the selected cluster, the sum of squared error (Definition 2) and the smallest 2 of the clusters through i times, then put these two clusters into cluster S. Repeating until generated K clusters.

In the two point, the algorithm find out the minimum sum of square errors in clustering results by using K-means clustering algorithm several times. Finally, Set the result as the initial center of CK-means clustering algorithm. The algorithm adopted the center which generated through two points search in optimization compared with the classical K-Means clustering algorithm. Therefore, it avoids the local optimization results obtained from random generation centers. At the same time, due to the “extreme point” principle, it can effectively reduce the number of clusters and improve the clustering efficiency.

1. The improvement of K-means based on the two - point search idea, and the combination of the “internal error square” and “the limit point principle”

Because of the K-means (V, K) algorithm has a large amount of computation in search of the target cluster and the determination of the target cluster, this paper divide the k-means algorithm into two steps, and implemented k-means by using the idea of the application of distributed computing and by a natural order based on Mapreduce programming framework.

The first process is to find the target cluster algorithm based on the MapReduce programming framework parallel implementation. The main principle is to calculate SSE_i of every node in the Map stage and find out the largest value of SSE_i and set it as target cluster.

The second step is to determine the limit point algorithm in the target cluster and implement the MapReduce programming framework. The main principle is that the cluster is divided into different nodes Mapper in Map phase, and then each node computes the distance from each point of the current node to each point of the cluster, and then distributes it to the Reduce; the Reducer stage get limit point by sorting these distances.

4 Experiment and Result Analysis

The experimental environment used in this paper is as follows: 5 sets of computers are used to build clusters. Configuration environment is as follows:

The above machines' hardware is consistent and configured as follows: I7 2.5 GHz 8 G memory, the operating system is: 14.04LTS Ubuntu (Table 1).

Table 1. Configuration list

IP	Name of nodes	Name of HDFS
192.168.1.101	Master	Namenode
192.168.1.102	Slave1	Datanode
192.168.1.103	Slave2	Datanode
192.168.1.104	Slave2	Datanode
192.168.1.105	Slave4	Datanode

4.1 CK-Means Algorithm Result Analysis

The experimental data sets are selected from UCI Machine Learning Repository, we also selected two data sets which are Breast Cancer sets and Synthetic Control Chart Time Series sets. The official description of the Cancer Breast data set has 286 samples, each of which has 9 attributes. Synthetic Control Chart Time Series data sets has 600 samples, however, how many attributes are not marked in each sample.

Table 2. The test results of the algorithm of the paper [13] and our algorithm in the cancer breast data sets

PNodes	CBK-means		Proposed algorithm	
	Accuracy/%	SSE_{min}	Accuracy/%	SSE_{min}
1	76	526	79	510
2	79	527	82	515
3	83	534	86	520
4	78	544	80	535
5	82	550	84	540

Tables 2 and 3 showed the experimental results of the accuracy of clustering results and average value of the minimum sum of square error respectively which based on Breast Cancel data sets and synthetic Control Chart Time Series data sets. The two comparison are corresponding to the algorithm of the paper [13] and this algorithm.

The CK-means algorithm, which is relative to the literature [13], determines the optimal number of clusters by improving the error square of the cluster and the principle of the limit point, and thus obtains higher accuracy of clustering and lower $SSE_{i\mathrm{min}}$. Additional, it can be seen that, with the increase of the number of nodes, the computing advantages of the clusters are more obvious, and the Hadoop can be extended to ensure the high reliability of the program. Therefore, the clustering

Table 3. The test results of the algorithm of the paper [13] and our algorithm in synthetic data sets

Nodes	CBK-means		Proposed algorithm	
	Accuracy/%	$SSE_{\min}$	Accuracy/%	$SSE_{\min}$
1	80	711 762	81	689 862
2	81	716 832	82	691 962
3	83	722 698	84	697 700
4	81	726 926	81	706 826
5	82	730 982	84	710 740

accuracy of the CBK-means algorithm has no obvious changed compared CK-means algorithm, but the value of $SSE_{i\min}$ is increasing.

4.2 Analysis of Data Spreading Rate

From Figs. 2 and 3, it can be seen that the expansion rate of the algorithm is gradually reduced when the number of nodes and the size of the test data sets is increasing, which is because the number of nodes increases the communication cost between nodes. The more bigger the size of the data sets, the CBK-means parallel algorithm and CK-means parallel algorithm of paper [13] of expansion rate are better, and the expansion efficiency curve will fall more smoothly. The two parallel algorithms have good extendibility in large data sets.

Experiments show that the CK-Means parallel algorithm compared to the CBK-Means parallel algorithm of paper [13], which is suitable for the cloud computing platform running on large scale data sets and have different degrees of improvement in clustering accuracy, speed ratio, expansion rate, etc.

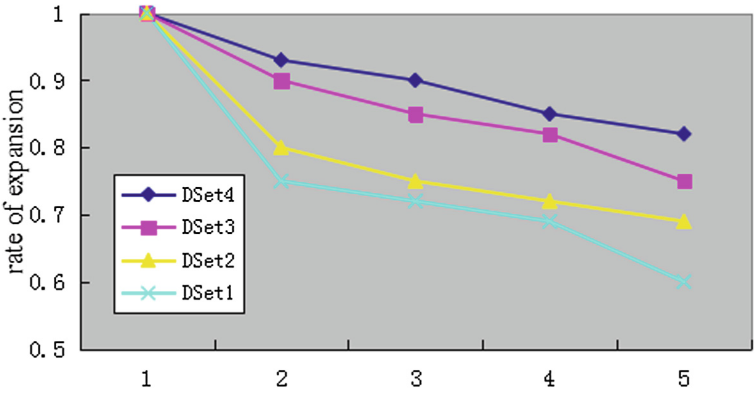

Fig. 2. The expression rate of CBK-means of 4 data sets in paper [13]

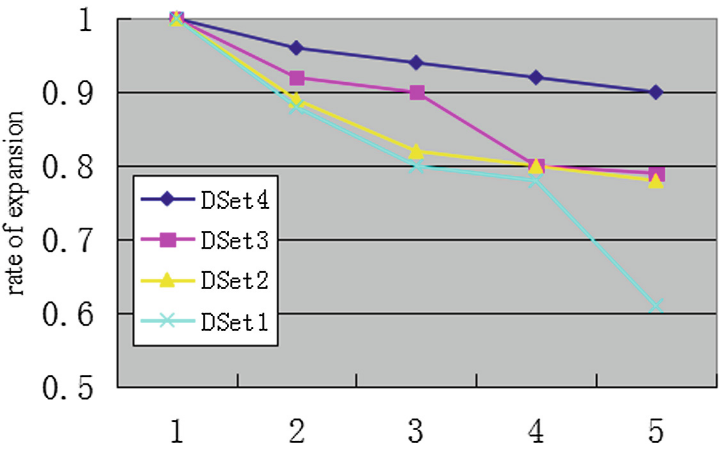

Fig. 3. The expression rate of CK-means of 4 data sets

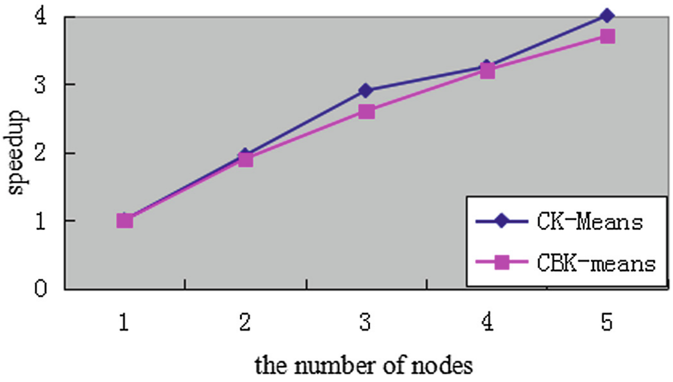

Fig. 4. The comparison of speed-up ratio of Dset1 data set

4.3 Analysis of Acceleration Ratio

In this experiment, we also follow the paper [13] using WEKA data generator RDGl to generate 4 more large data sets, the data set randomly which number was 4000, 8000, 16000, 32000 etc., whose name are DSet1, DSet2, DSet3, DSet4.

From Figs. 4 and 7 it can be seen that, the growth of acceleration rate of vertical axis tends to slow down with the increase in the number of nodes in the transverse, mainly because of the increase of the nodes lead to the cost of communication overhead is gradually increasing. The speedup of the two parallel algorithms is more close to the linear increase for the larger data sets. When the size of the data set is larger, the CK-means parallel algorithm is significantly higher than that of the CBK-means parallel algorithm (Figs. 5 and 6).

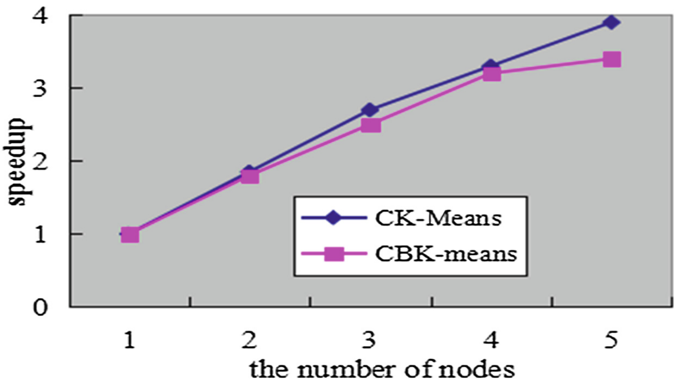

Fig. 5. The comparison of speed-up ratio of Dset2 data sets

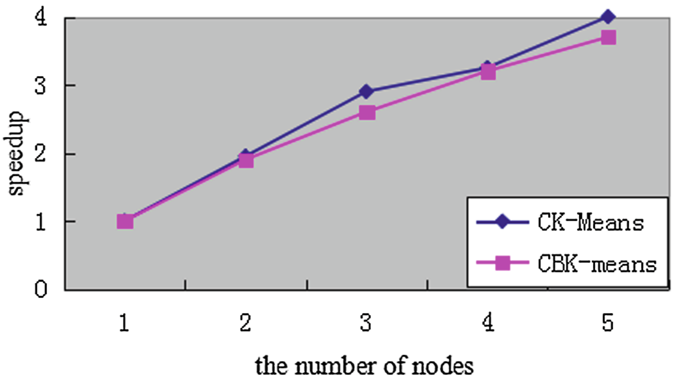

Fig. 6. The comparison of speed-up ratio of Dset3 data sets

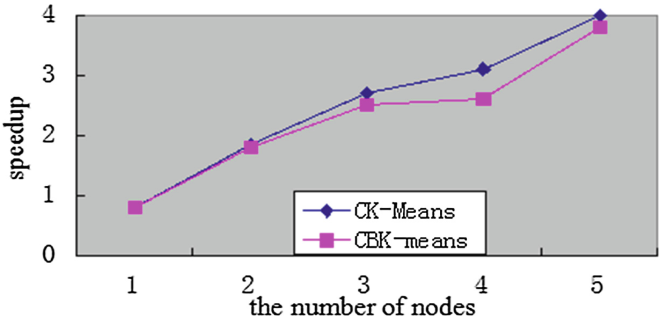

Fig. 7. The comparison of speed-up ratio of Dset3 data sets

5 Conclusion

This paper designed and implemented a CK-means algorithm based on MapReduce parallel computing model of the Hadoop platform. Through the UCI data set test, the results show that the larger the data, the more the number of cluster nodes, the better the effect of the algorithm, the higher the efficiency of clustering, and can be effectively applied to the mining of massive data. At the same time, it also proves that MapReduce can effectively improve the data processing capability of large data.

Acknowledgments. This research is supported by the Fundamental Research Funds for the Central Universities (No. 2015ZM039) and scientific research project for Guangdong University of Science and Technology (No. GKY-2014KYYB-10).

References

1. Qian, W.N., Zhou, A.Y.: Analyzing popular clustering algorithms from different viewpoints. J. Softw. **13**(8), 1382–1394 (2002)
2. Gustavo, E.A., Batista, P.A., Monard, M.C.: Annalysis of four missing data treatment methods for supervised learning. Appl. Artif. Intell. **13**(5/6), 519–533 (2003)
3. Bao, L., Li, Q.: Combat Big Data. Tsinghua University Press, Beijing (2014)
4. Wen, C.: Parallel Clustering Algorithm Based on MapReduce. Zhejiang University, HangZhou (2011)
5. Jiang, X., Li, C.: Parallel implementing k-means clustering algorithm using MapReduce. J. Huazhong Univ. Sci. Tech. (Nat. Sci. Ed.) **39**(1), 120–124 (2011)
6. Li, Y.: Research on parallelization of clustering algorithm based on MapReduce. Sun Yat-sen University, Guangzhou
7. Xue, S.-J., Pan, W.: Parallel Pk-means algorithm on meteorological data using MapReduce. J. Wuhan Univ. Technol. **34**(12), 139–142 (2012)
8. Ji, S.-Q., Shi, H.-B.: K-means clustering ensemble based on MapReduce. Comput. Eng. **39** (9), 84–87 (2013)
9. Xie, X., Li, L.: Reseach on parallel k-means algorithm based on cloud computing platform. Comput. Meas. Control **22**(5), 1510–1512 (2014)
10. Zhang, X., Zhang, G., Liu, P.: Improved k-means algorithm based on clustering criterion function. Comput. Eng. Appl. **47**(11), 123–128 (2011)
11. Su, M.C., Chou, C.H.: A modified version of the k-means algorithm with a distance based on cluster symmetry. IEEE Trans. Pattern Anal. Mach. Intell. **23**(6), 674–680 (2001)
12. Fu, N., Qiao, L.Y., Peng, X.Y.: Blind recovery of mixing matrix with sparse sources based on improved k-means clustering and hough transform. Chin. J. Electron. **37**(4), 92–96 (2009). (Ch)
13. Gao, R., Li, J., Xiao, Y., Zhu, S., Peng, W.: Parallel algorithm based on K-means clustering in cloud environment. J. Wuhan. Univ. (Nat. Sci. Ed.) **61**(4), 368–374 (2015)