International Core Journal of Engineering Volume 7 Issue 4, 2021
ISSN: 2414-1895 DOI: 10.6919/ICJE.202104_7(4).0062

Research on Canopy-Kmeans Algorithm based on Hadoop

Chaoju Hu[1,a], Jiaxing Zhai[1,b]

[1]School of Control and Computer Engineering, North China Electric Power University, Baoding 071000, China.

[a]huchaoju@163.com, [b]zhaijx95@163.com

Abstract

This paper analyzes the advantages and disadvantages of traditional K-means and Canopy algorithms, and proposes an improved K-means algorithm based on Canopy. At the same time, it uses the "min-max principle" to improve its space complexity and randomness problems, and applies it to the MapReduce programming model under the Hadoop platform. Experiments show that this method is more accurate and accurate than the traditional K-means and Canopy algorithms. stability.

Keywords

Hadoop; MapReduce; Canopy; K-means Algorithm; Clustering.

1. Introduction

Clustering is to divide the data set into several categories or clusters according to the idea of "things are clustered together", so that the data in each cluster is similar to the greatest extent, and belongs to an unsupervised learning process [1]. At present, most clustering algorithms are suitable for centralized data processing, and the efficiency of the algorithms is limited by the processing capacity of a single machine. For this reason, existing clustering algorithms all have poor scalability. Therefore, distributed clustering algorithms in which multiple computers participate in the calculation have become the focus of current clustering algorithms [2]. At present, in the existing distributed clustering algorithm research, the K-means algorithm is compared with the density-based DBDC algorithm, the hierarchical clustering-based RACHET algorithm, the CHC algorithm, and the CPCA clustering algorithm based on comprehensive principal component analysis. It has the advantages of simple algorithm implementation and high computational efficiency, so it has become the most widely used distributed clustering algorithm. Canopy-Kmeans is a clustering algorithm optimized for K-means. After Canopy is introduced, it only compares the distance between the object and the center point in the same area each time, and greatly reduces the running time of the entire clustering by reducing the number of comparisons. Improve the calculation efficiency of the algorithm. In the actual application of the algorithm, the initial seed point of the cluster (initial Canopy center point), the number of clusters k (the number of Canopy), the size of the Canopy area, and other initial values need to be set in advance. The selection of the initial value will affect the final clustering. The quality has a great impact [5]. For example, Canopy center point selection is too dense, which may cause the algorithm to fall into the local optimum. The Canopy area radius directly affects the execution efficiency and classification accuracy of the algorithm. The initial value setting of the Canopy-Kmeans algorithm is generally based on experience or It has been tried many times, so it has greater blindness and randomness. In order to solve this problem, this paper introduces the "minimum maximum principle" to improve the algorithm, proposes an optimized selection and setting method of Canopy center point, and uses the MapReduce parallel computing framework proposed by Google to implement the algorithm[4], and at the same time uses distributed storage Massive Internet news data is used as the background of clustering application, which verifies the practical application value of the improved algorithm[8].

ISSN: 2414-1895 DOI: 10.6919/ICJE.202104_7(4).0062

2. Algorithm idea

2.1 K-means algorithm idea

K-means is a classic partition-based clustering method. Its basic idea is: first select K centroids from the initial data set, and treat these k points as the center points of k clusters. Then calculate the distance from each remaining point in the space to each center point, and cluster each point into the cluster closest to the point. Finally, calculate the average of all sample coordinates in each cluster, and use this average as the new center point. By continuously repeating the above process, the cluster center will eventually converge or meet the requirements. The advantages of this algorithm are simple and efficient, but the disadvantages are also obvious: (1) From its principle, it can be seen that the results of the K-means algorithm are closely related to the K value. When there is a lot of data, it is often not known how many types are included, And the result of artificially prescribed K value is often biased [12].

2) The algorithm needs to select k data from the data set as the initial center point, and then perform subsequent clustering based on the selection, and then determine the result through continuous iteration, so the choice of the initial center point has a great impact on the final result Impact.

(3) Sensitive to individual outlier data, and these data have a great influence on the final result.

(4) The algorithm needs to constantly update the center point, which leads to an increase in the number of iterations when there are more data, and an increase in overhead, which affects efficiency [14].

2.2 Canopy algorithm idea

Canopy algorithm is a simple and fast clustering analysis method. It does not need to divide the center point in advance, thus avoiding the problems encountered by the K-means algorithm regarding the selection of initial values. This algorithm is often used to assist other algorithms. The data set is roughly divided into clusters, and the divided data is then handed over to other algorithms for fine clustering.

The basic idea of the Canopy algorithm is: First, set T1 and T2 as loose threshold and tight threshold respectively, and T1>T2, randomly select a data from the data set as the center point. Then use some known distance calculation methods to calculate the distance between the center point and other data in the data set, and weakly mark the data with a distance less than T2. These points will not become new center points because they are closer to the center point. For points whose distance is less than T1, add them to the set of Canopy center points. Then select the unlabeled data from the original data set as the new Canopy center and repeat the subsequent process until all the data in the data set are labeled.

From this we can see that the Canopy clustering results are related to the division of T1 and T2 values, and there may be duplicate subsets in the clustering results, which reduces the impact of outliers in the K-means algorithm and increases fault tolerance , And the classification result can be passed to the K-means algorithm as the K value, which in turn weakens the K-means algorithm's dependence on the selection of the K value. However, the threshold of the Canopy algorithm still needs to be manually determined. If T1 is too large, the same data may belong to multiple Canopy collections. If the T2 value is too large, the number of clustering results may be too small, so how to determine T1 and T2 have become the key points of this algorithm.

3. Algorithm improvement

In order to solve the above situation, the distance between the initial center points of the Canopy algorithm should be set as far as possible. This paper uses the principle of minimum and maximum to improve the Canopy algorithm to determine the initial center point of Canopy clustering. The basic idea is as follows:

(1) First, select a point in the data set C = {x1,x2,...,xi} as the first center point xA randomly;

ISSN: 2414-1895 DOI: 10.6919/ICJE.202104_7(4).0062

(2) Calculate the minimum distance from the remaining data points in the data set to xA, and select the point with the largest distance from it as the second center point xB;

(3) Continue to calculate the minimum distance min{ d(xA, xB)} from the remaining data points in the data set to xA, xB, and select the largest of these minimum distances max{ min [d(xA, xB)] } as the third center Point xC. By analogy, in the end, all initial Canopy center points {xA, xB,..., xK} should meet the conditions:

$$DistList = \min \{distance(Xn+1,Xi), i=1,2,...n) \}$$

$$DistMin(n+1)=\max\{DistList\}$$

In the formula, DistList represents the minimum distance between the n+1 center point and the first n center points, and DistMin(n+1) means that the n+1 center point should be the largest of all the shortest distances. The Canopy center point selected by this method can effectively avoid the problem of selecting the T2 threshold. This method of selecting the center point will show the following rules in the application: when the number of center points is lower than or greater than the number of true center points DistList(n+1) has a small degree of change. When the number of center points is close to the number of true center points, DistMin(n+1) changes greatly. In order to determine the optimal number of center points and the value of T1, depth is introduced The index Depth(i) represents the magnitude of change, which is defined as the formula:

$$Depth(k)=|DistMin(k)-DistMin(k-1)| + |Dist(k+1)-DistMin(k)|$$

When k is close to the number of real clusters, Depth(K) achieves the maximum value. At this time, setting T1=DistMin(K) makes the result optimal. The improved Canopy algorithm based on this method solves the problem of setting the T1 and T2 thresholds, and also solves the problem of Canopy center point selection, which can provide a more stable set for the K-means algorithm, making the K-means algorithm more efficient and accurate [10].

3.1 Algorithm implementation steps

The implementation steps of Canopy-K-means clustering algorithm are as follows:

(1) Read data characteristics.

(2) Steps to execute Canopy algorithm to obtain initial cluster centers:

(I) Store the characteristic matrix of the data in a List and determine the distance threshold T1 based on the "maximum and minimum principle".

(II) Randomly take a vector P from the List and calculate the distance between P and all Canopy. If there is no Canopy yet, take P as the first Canopy. If the distance between P and a Canopy is within T1, then point P Join this Canopy.

(III) If the distance between vector P and a certain Canopy is within T2, delete P from the List.

(IV) Repeat steps 2 and 3 until the List is empty.

(3) Execute the K-means algorithm to calculate the distance between the data object and the cluster center, the steps are as follows:

(I) Select the centers of K clusters obtained after the execution of the Canopy algorithm as the initial cluster centers.

(II) Traverse all the remaining vectors to calculate the distance to the K centers, and classify them into the cluster where the nearest center point is located.

(III) Recalculate the center point of each cluster for the calculated K clusters.

(IV) Calculate the distance from each point to K center points again, and classify it into the cluster where the closest point is located.

(V) Repeat steps 2 to 4 until the position of the center point no longer changes or reaches the predetermined number of iterations [11].

(4) Save the clustering results.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202104_7(4).0062

3.2 Algorithm for parallel optimization

Under the MapReduce programming framework, the algorithm can be optimized in parallel. The steps are as follows:

(1) The data fragments stored in HDFS are delivered to the Map node.

(2) Each Map node reads the data that needs to be processed, and enters it in the form of <key, value> key-value pairs. For example, key is the user number UID, value is the page stay time StayTime, and each Map node is For the data set you want to process, use the "maximum minimum principle" to determine the Canopy center point, the output key is the Canopy number, and the value is the Canopy center point [13].

(3) A large amount of intermediate data will be generated after the processing of the Map phase, and it will cause a large network bandwidth overhead to be directly transmitted to the Reduce node. In order to reduce the communication burden between nodes, the intermediate key-value pairs processed by the Map node can be merged, that is, the key-value pairs with the same key are merged into a set of values under the same key. After obtaining the partial merge result, pass it into the Reduce function [6].

(4) Reduce receives the output value of the Map, and then calculates the Canopy center point to obtain the Canopy center point of the entire data set. The process of canopy algorithm mining is shown in Figure 1.

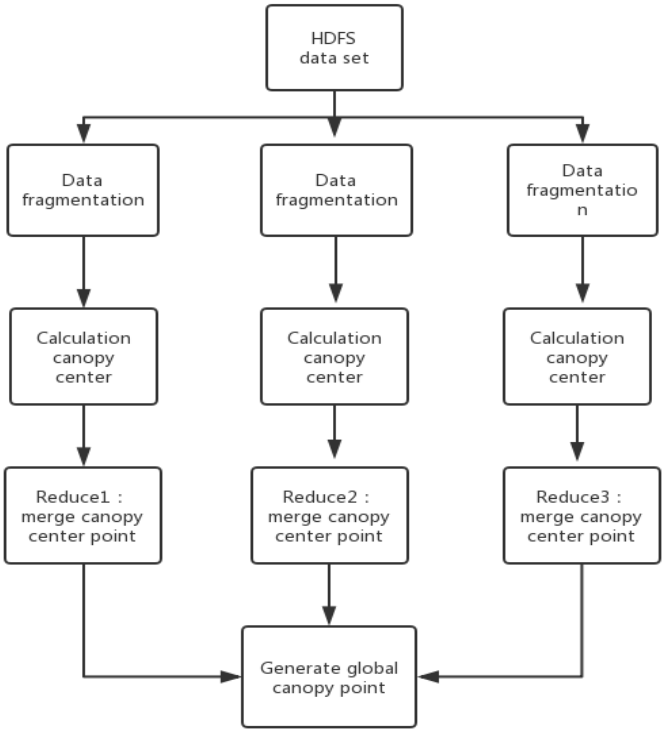

Figure 1. Canopy algorithm flowchart

(5) Send n Canopy to n Map, use the obtained Canopy center point as the initial cluster center of K-means clustering, use K-means to cluster each Canopy data, and output the key as the cluster center number, Value is the center point of the cluster.

(6) The Combiner receives the output value of the Map, merges the clusters with the same key value and sends it to Reduce.

(7) Reduce updates the cluster center, obtains the global cluster center, and divides the data into each cluster [7].

ISSN: 2414-1895 DOI: 10.6919/ICJE.202104_7(4).0062

(8) Repeat the process 5-8 until the final requirement is reached and output the information of K clusters as the clustering result.

The process of k-means algorithm mining is shown in Figure 2.

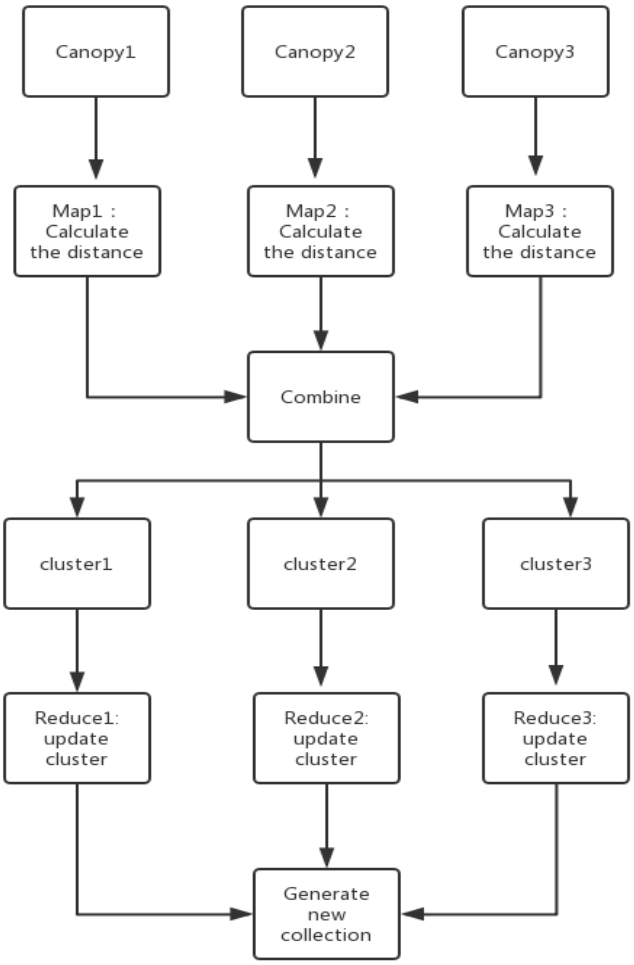

Figure 2. K-means algorithm flowchart

3.3 Center point selection optimization

The Canopy optimization selection method based on the “minimum-maximum principle” is very time-consuming in practical applications, especially for processing distributed storage of massive data. Therefore, in order to improve the execution efficiency of the algorithm and shorten the running time of the algorithm, the execution process of the algorithm can be optimized. Considering that the initially selected Canopy center point is not the final cluster center point (it needs to be dynamically adjusted in the K-means iteration stage), in the Canopy preliminary selection stage, it is only necessary to ensure that the distance between the initial Canopy center points is the largest. The optimization strategy of the algorithm is:

(1) The data point with the closest and farthest distance from the origin of the coordinates is used to replace the data point with the longest initial distance in the data set to avoid solving in the global scope. The execution process of the algorithm is shown in Figure 3.

(2) For distributed storage of massive data sets, first generate local candidate Canopy center points in each node based on the "minimum maximum principle", and then obtain the global Canopy center points based on this strategy, instead of directly in the global data Find the Canopy center point in space.

ISSN: 2414-1895 DOI: 10.6919/ICJE.202104_7(4).0062

(3) When each node generates a local candidate Canopy center point, in order to reduce the number of iterations, the program iterates to the end of √N (N is the data size of the node).

4. Experiments and results

4.1 Algorithm strategy comparison

From the analysis in Table 1, it can be seen that in the Canopy center point generation stage, the former takes a long time to determine the optimal number of clustering categories, and its running time is proportional to the data size, while the latter randomly sets the area radius so that There are fewer iterations, so the former is less efficient than the latter at this stage. In the K-Means iteration stage, the runtime of the algorithm There is a positive correlation between the number of iterations and the radius of the Canopy area. When the Canopy area radius of the former is larger, the iterative calculation process of the center point is relatively complicated with the increase of data volume. Therefore, the latter performs better in the number of iterations and algorithm efficiency at this stage. former. However, considering that the random selection of the Canopy strategy generally requires multiple trials to obtain a better classification result, the Canopy-Kmeans algorithm is significantly better than the random selection of the Canopy strategy from the overall time of cluster classification.

Table 1. Canopy - Kmeans algorithm and Canopy strategy result comparison

Items	Canopy-Kmeans	Canopy
Original cluster vector size	100MB	100MB
The number of canopy	653	2527
number of canopy iterations	172	28
number of kmeans iterations	50	18
Center point generation time	45min	4min
Iteration generation time	140min	80min

4.2 Comparison of calculation modes

Comparison of MapReduce parallel computing mode and stand-alone mode. In order to compare the performance difference between the improved algorithm in processing massive data in distributed storage and in single-machine centralized mode processing large-scale data, in the experiment For the same data set, two models were used for comparative analysis. The experimental results are shown in Table 2. From the analysis in Table 2, it can be seen that the good scalability of Hadoop enables it to cope with the continuous expansion of the data scale and ensures the high reliability of the program. It can be seen that the Canopy-Kmeans algorithm based on the "minimum maximum principle" optimized selection of the Canopy method can not only ensure the execution efficiency of the algorithm, but also improve the accuracy of classification under the MapReduce parallel computing framework. Therefore, the parallel expansion of the algorithm has strong application value.

Table 2. Comparison of stand-alone and hadoop platform

category	Stand-alone	hadoop
Number of articles	200000	200000
The amount of data	30MB	30MB
Number of nodes	1	200
Execution time	10h	3h

5. Conclusion

Taking the massive Internet news data clustering as the application background, using the inherent parallelism of the MapReduce framework, the parallel extension and application of the Canopy-

ISSN: 2414-1895 DOI: 10.6919/ICJE.202104_7(4).0062

Kmeans algorithm are studied and explored, and the blindness of Canopy selection is solved based on the "minimum maximum principle" And arbitrariness, the experimental results show: based on the "minimum maximum principle" optimal selection Compared with the random selection of Canopy strategy, the Canopy method improves the overall running time of clustering, the accuracy of classification, and the "anti-noise" ability. Moreover, the implementation of the algorithm based on MapReduce makes the program not affected by the data scale and ensures the algorithm's performance. High reliability. However, since the generation of Canopy center points is relatively time-consuming, on the basis of not affecting the accuracy of algorithm classification, how to further improve the efficiency of generating Canopy center points by the “minimum-max principle” requires further research.

References

[1] Yongjun, ang, Jing Sun. The Research of Meteorological Data Mining Using Discrete Bayesian Networks Classifier Based on Hadoop[P]. Proceedings of the 2015 International Conference on Electrical, Computer Engineering and Electronics, 2015.

[2] Khabat Khosravi, Prasad Daggupati, Mohammad Taghi Alami, Salih Muhammad Awadh, Mazen, Ismaeel Ghareb, Mehdi Panahi, BinhThai Pham, Fatemeh Rezaie, Chongchong Qi, Zaher Mundher Yaseen. Meteorological data mining and hybrid data-intelligence models for reference evaporation simulation: A case study in Iraq[J]. Computers and Electronics in Agriculture,2019,167.

[3] Data Mining; Reports on Data Mining Findings from Edith Cowan University Provide New Insights (A combination of meteorological and satellite-based drought indices in a better drought assessment and forecasting in Northeast Thailand) [J]. Computers, Networks & Communications,2015.

[4] Dan Meng, Jizhong Han, Jianfeng Zhan, Bibo Tu, Xiaofeng Shi and Le Wan, "Transformer: A New Paradigm for Building DataParallel Programming Models", Vol. 30, Issue 4, pp. 55-64, 2010.

[5] Dawei Jiang, Antony K.H. Tung and Gang Chen, “MAP-JOINREDUCE: Toward Scalable and Efficient Data Analysis on Large Clusters”, IEEE Trans. Knowledge and Data Engineering, Vol. 23, No. 9, pp. 1299-1311, September 2011.

[6] Wei Qu. Efficient File Accessing Techniques on Hadoop Distributed File Systems[A]. ICYCSEE Steering Committee. Abstract of the Second International Conference of Young Computer Scientists, Engineers and Educators, ICYCSEE 2016 PartI[C].

[7] Zhang C C, Zhang H Y, Luo J C, et al. Massive data analysis of power utilization based on improved K-means algorithm and cloud computing[J]. Journal of Computer Applications, 2018,38(1):159-164.(in Chinese).

[8] Lu S Y, Wang J Y, Zhang X L, et al. Optimization algorithm of K-means clustering based on Hadoop[J]. Journal of Inner Mongolia University of Science and Technology, 2016,35(3):264-268.(in Chinese).

[9] HTML: https://hadoop.apache.org/docs/stable/hadoop-project-dist/hadoop-common/SingleCluster.html.

[10] G.s. Bhathal and A.S. Dhiman, "Big Data Solution: Improvised Distributions Framework of Hadoop," 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2018, pp.35-38. doi: 10.1109/ICCONS.2018.8663142.

[11] Cui X, Zhu P, Yang X, et al. Optimized big data K-means clustering using MapReduce[J]. Journal of Supercomputing. 2014,70(3):1249-1259.

[12] Mao D H. Improved Canopy-Kmeans algorithm based on MapReduce[J]. Computer Engineeringand Applications, 2012,48(27):22-26.(in Chinese).

[13] Tao Y, Yang F, Liu Y, et al. Research and optimization of K-means clustering algorithm[J]. Computer Technology and De- velopment, 2018,28(6):90-92.(in Chinese).

[14] Dean J, Ghemawat S. MapReduce: Simplified data processing on large clusters[J]. Communications of the Association for Computing Machinery, 2008,51(1):107-113.