UDC 004.75

АНАЛИЗ k - СРЕДНИХ АЛГОРИТМА В ТЕХНОЛОГИЙ HADOOP И MPJ EXPRESS.

Темирбекова Ж.Е., Тюлепбердинова Г.А.

Казахстан, г. Алматы, Казахский Национальный университет им. аль-Фараби

zhanerke_3089@mail.ru

Один из наиболее популярных методов кластеризации является алгоритм k means, из-за его легкой реализации, простоте, эффективности и эмпирических успехов. Целью данного исследования является проведение экспериментов алгоритма k means в технологии Hadoop и реализовать параллельный алгоритм на языке Java с обращениями к библиотеке MPJExpress.

Ключевые слова: кластеризация, оптимизация параллельных вычислений, распределенная программа.

ANALYZE k -MEANS ALGORITHM IN TECHNOLOGY HADOOP AND MPJ EXPRESS.

Temirbekova Zh.E., Tyulepberdinova G. A.

The most famous clustering algorithm is k means because of its easy implementation, simplicity, efficiency and empirical success. The goal of this study is to perform k means clustering using Hadoop and implement a parallel algorithm in Java with calls library MPJ Express.

Keywords: clustering, optimization of parallel computing, distributed program.

Clustering is one of the most popular methods for exploratory data analysis, which is prevalent in many disciplines such as image segmentation, bioinformatics, pattern recognition and statistics etc.

Images obtained using space remote sensing of the Earth play a crucial role in research, industrial, economic, military and other applications. Development of remote sensing spacecraft and associated ground-based imaging actively conducted throughout the world [1]. For the analysis of hyperspectral remote sensing images, there are many algorithms. One of the most popular methods of clustering algorithm is k -means.

Algorithm k -means

The basic idea of k -means algorithm is to minimize the distances between objects in a cluster. Stop computing occurs when minimizing the distance reaches a certain threshold. Minimized function is as follows: $J = \sum_{k=1}^{M} \sum_{i=1}^{N} d^2(x_i, c_k)$, where $x_i \in X$ – object clustering, $c_j \in C$ – center of the cluster. $|X| = N, |C| = M$. At the time of the start of the algorithm must be known by C (number of clusters). Select the number may be based on the results of previous studies, theoretical considerations or intuition [2].

Parallelization algorithm k-means

k-means algorithm can be run on very large data sets, the order of hundreds of millions of points and tens of gigabytes of data. Because it works on such large data sets, and also because of the special characteristics of the algorithm, it is a good candidate for parallelization. In the course of calculation algorithms have been implemented in the form of serial and parallel programs on the Java programming language using the technology MPI. On a multiprocessor computer Mechanics and Mathematics Faculty KazNU calculations were carried out for a parallel algorithm.

Clustering algorithm k-means in MapReduce

MapReduce is a programming model and appropriate technology for processing large data sets. MapReduce divides the input data set into independent parts. Processing takes place in two stages: using valve functions Map and gearboxes Reduce [3].

The algorithm works iteratively in several stages, in the following manner:

1. In the first stage, Mappers reads share input and compresses the original data set into a smaller set of data, the so-called auxiliary cluster. These auxiliary clusters help to present raw data in case of a limited amount of RAM.

2. Each Mapper creates k initial cluster of these auxiliary clusters, which are then sent to the Reducer.

3. Reduce combines clusters from each Mapper and recalculates the centroids of k clusters.

4. The centers of gravity at the moment thus returned to the original broadcast by Mapper operations.

5. Now everyone can use Mapper new centroids and reassign its subsidiary centers of gravity of these clusters. Mapper send its local clusters back to the Reducer.

6. Reducer again combines clusters and recalculates the centroid.

7. This procedure is repeated until Reducer decides to stop repeated data Mapper. This typically occurs when the algorithm converges.

The work was implemented distributed clustering algorithm k-means using the technology of MapReduce.

Map function:

(global object, in_key, in_value), global object contains the initial clustering centers, in_key has no usefulness, in_value is a string like (pixel_id, R, G, B). Output: (out_key, out_value), out_key is a string represents a clustering center, out_value is a same string as in_value.

```
1: construct initial clustering centers Array from global object;
2: labPixel = parseString ( in_value );
3: minDistance = MAX_VALUE;
4:initial_array_subscript = -1;
```

5: for (j = 0; j< Array.length; ++j) {

6: dist = cal_dist_labpixel_to_centers(labPixel, Array[j]);

if (dist < minDistance) { minDistance = dist; initial_array_subscript = j; } }

7: out_key = Array[initial_array_subscript];

8: out_value = in_value;

9: writeToHDFS(out_key,out_value);

10: output(out_key,out_value);

11: End;

Reduce function:

Reduce function Input: (in_key, in_value), in_key is a string represents a clustering center, in_value is a string like (pixel_id, R, G, B).

Output: (out_key, out_ value), out_key is a string represents the number of values which have the same key in iterator, out_value is a string represents a new clustering center after adjustment.

1: set the initial value of counter to 0;

2: set temp_ave like (null,null,null,null);

3: while(in_key.hasNext()) {

4: temp_ave=temp_ave+abs(in_value.Next() - temp_ave)/(counter + 1);

++counter; }

5: out_key = counter.ToString();

6: out_value = temp_ave;

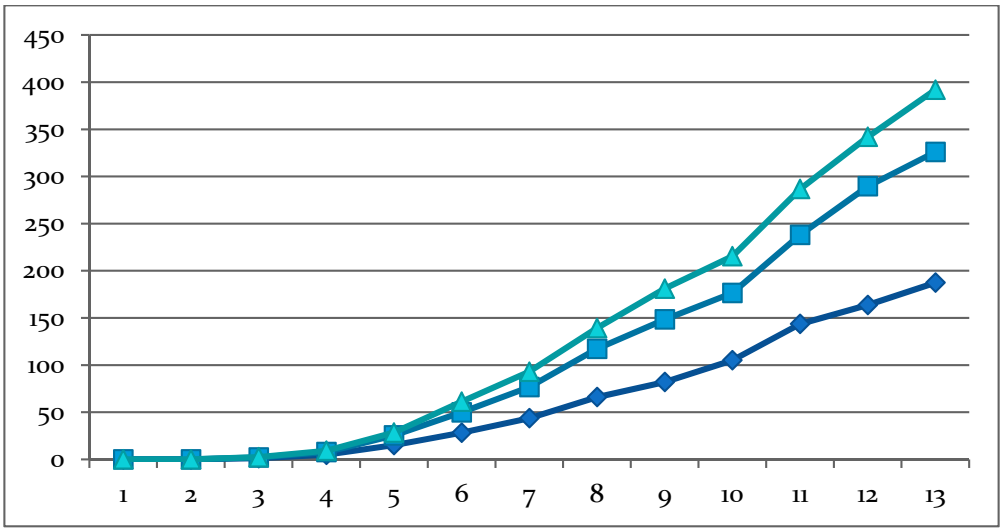
7: output(out_key,out_value);

8: End; [4].

Thus, k-means algorithm is well parallelizable. Application of MPI and MapReduce technologies provides a significant acceleration compared to the implementation of the non-parallel algorithm.

Table 1 – Java software using library the MPJ Express and Hadoop technology.

N value of the points	time (Ts, sec) sequential	time (Ts, sec) parallel	technology Hadoop
50	0	0.062	0.028
100	0.035	0.031	0.0154
500	1.321	0.781	0.2952
1000	4.924	2.812	1.552
2000	15.264	10.261	2.9995
3000	28.345	21.547	11.399
4000	43.78	32.953	16.32
5000	66.155	61.188	21.967
6000	82.06	66.375	32.617
7000	105.21	71.312	38.949
8000	143.671	94.484	48.579
9000	168.82	125.94	52.313



Picture 1- Hadoop, parallel, sequential k -means

In conclusion, MapReduce paradigm significantly reduce time image processing algorithm. Our calculations experimented using the platform for distributed computations Hadoop MapReduce paradigm. Hadoop platform allowed us to change the scope of the report's calculations using multiple computing nodes tally.

Literature

1. Кашкин В.Б., Сухинин А.И. Дистанционное зондирование Земли из космоса. Цифровая обработка изображений: Учебное пособие. – М.: Логос, 2001г. 264 стр.

2. Р. Миллер, Л. Боксер. Последовательные и параллельные алгоритмы. ИздательствоБином. Лаборатория знаний 2006г., 408стр.

3. J. Dean, S. Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. Communications of The ACM, 2008. 51(1), 107-113.

4. W. Zhao, H. Ma, Q. He, “Parallel K-Means Clustering Based on MapReduce,” Cloud Computing, vol. 5931, 2009. pp. 674-679.