Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 107 (2017) 442 – 447

Procedia Computer Science

International Congress of Information and Communication Technology (ICICT 2017)

Parallel Implementation of Density Peaks Clustering Algorithm Based on Spark

Rui Liu[a], Xiaoge Li[a*], Liping Du[a], Shuting Zhi[a], Mian Wei[b]

[a]School of Computing, Xi'an University of Posts and Telecommunications, Xi'an 710121, China
[b]Tulane University, New Orleans, LA 70118, USA
** Corresponding author: lixg@xupt.edu.cn Tel.: 15055114114*

Abstract

Clustering algorithm is widely used in data mining. It attempt to classify elements into several clusters, and the elements in the same cluster are more similar to each other meanwhile the elements belonging to other clusters are not similar. The recently published density peaks clustering algorithm can overcome the disadvantage of the distance-based algorithm that can only find clusters of nearly-circular shapes, instead it can discover clusters of arbitrary shapes and it is insensitive to noise data. However it needs calculate distances between all pairs of data points and is not scalable to the big data, in order to reduce the computational cost of the algorithm we propose an efficient distributed density peaks clustering algorithm based on Spark's GraphX. This paper proves the effectiveness of the method based on two different data set. The experimental results show our system can improve the performance significantly (up to 10x) comparing to MapReduce implementation. We also evaluate our system expansibility and scalability.

Keywords: density peaks; clustering; Spark; GraphX; big data

1. Introduction

Clustering analysis is an important technique in machine learning and data mining. Clustering analysis[1] divides elements into several clusters, and the elements in the same cluster are more similar to each other meanwhile the elements belonging to other clusters are not similar. At present, there are many clustering algorithms, such as partition-based method(e.g. k-medoids[2], k-means[3]), hierarchical-based method(e.g. Agglomerative Nesting(AGNES)[4]), density-based method(e.g. Density-based Spatial Clustering of Applications with Noise(DBSCAN)[5]), grid-based method(e.g. a Grid-Clustering algorithm for High-dimensional very Large spatial databases(GCHL)[6]) and probability model based method. In 2014, a paper about density peaks clustering algorithm

Peer-review under responsibility of the scientific committee of the 7th International Congress of Information and Communication Technology
doi:10.1016/j.procs.2017.03.138

was published in Science magazine[7]. The core of the algorithm is that cluster centers are characterized by a higher density than their neighbors and by a relatively large distance from points with higher densities[7].

In this paper, we present a parallel implementation of density peaks clustering system using GraphX based on Spark. We study the effectiveness of the method and evaluate the running time under different number of nodes at the same amount of data or under different amount of data at the same number of nodes. Finally, we compare the running time of Spark and MapReduce to see which is better.

The rest of this paper is organized as follows. In Section 2, we review the density peaks clustering algorithm and Spark RDD model. In Section 3, we introduce our parallel density peaks clustering System based on Spark. Section 4 provides the details of our experiment and deeply analyzes the results. Finally, in Section Conclusions we conclude our contribution and indicates our directions for future research.

2. Related works

This section reviews the density peaks clustering algorithm and introduces Spark RDD model.

2.1. Density peaks clustering algorithm

The kernel parts of density peaks clustering algorithm are computing two value for point i : the local density ρ_i and the distance from points of higher density δ_i . And for point i , the local density ρ_i is defined as:

$$\rho_i = \sum_j \chi\left(d_{ij} - d_c\right) \tag{1}$$

Where $\chi(x)=0$ if $x \ge 0$ and $\chi(x)=1$ otherwise, and d_{ij} is the distance between point i and point j meanwhile d_c is a cutoff distance. Typically, to the point i , ρ_i is equal to the number of points that are closer than d_c . Remarkably, the algorithm is robust with respect to the choice of d_c for large data sets and the algorithm is sensitive only to the relative magnitude of ρ_i in different points.

δ_i is calculated by getting the minimum distance between the point i and any other point with higher density:

$$\delta_i = \min_{j:\rho_j > \rho_i}\left(d_{ij}\right) \tag{2}$$

For point i with highest density, we take $\delta_i = \max_j(d_{ij})$. And δ_i is much larger than the typical nearest neighbor distance only for points that are global or local maxima in the density. Therefore, cluster centers are regarded as points for which the value of δ_i is anomalously large.

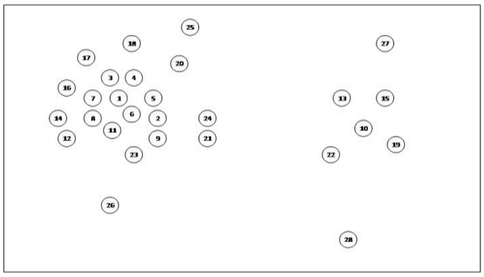

Fig. 1. Point distribution.

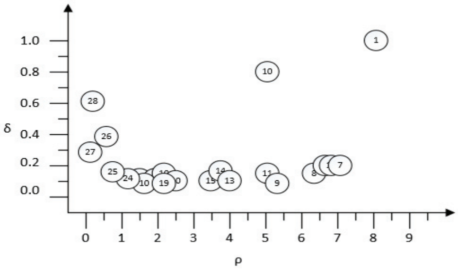

Fig. 2 Decision graph for the data in Fig. 1.

For each point i , ρ_i and δ_i could be expressed in a two-dimensional decision graph. For example, Fig. 1 shows 28 point embedded in a two-dimensional space, and points 1 and 10 are the density maxima, i.e. points 1 and 10 are