BIRCH CLUSTERING ALGORITHM

– Balanced Iterative Reducing and Clustering using Hierarchies

Created by Bîscă Emanuel (✉ emanuel.bisca@stud.ubbcluj.ro) on November 2020

MAIN IDEAS

In our society, with things evolving so fast, the data sets are growing larger and larger. As a result, the tools handling those data sets and the problem generated by them must evolve as well. Nowadays, the clustering algorithms are regaining a lot of attention, together with the rise of parallelized computing architecture development. Although the progress exists and it's easy observable, most of the existent clustering algorithms suffer from two major drawbacks: they do not scale well with the increasing dataset sizes and quite often they need proper parametrization which is usually difficult to provide.

In the literature is easy to found a lot of algorithms that are working to solve as many issues as there are, but we are never going to find the algorithm that does everything in the best way possible. In the paper, it is presented BIRCH clustering algorithm as one answer to a question asked by many pioneers interested in researching in this field.
The content of the paper is as it follows:

- An introduction to the researched topic, in which is presented the field of Hierarchical Clustering.
- The contributions of BIRCH and its limitations
- The background of mathematics that helps us understand the concepts needed when using BIRCH
- The concept of Clustering Feature together with the CF tree and why are they the main ideas behind BIRCH's incremental clustering.
- The steps that BIRCH is doing are explained extensively in the second chapter of the report
- Two major concepts of BIRCH clustering algorithm are presented to the reader: the reducibility, here we discuss about the Reducibility Theorem; and the Threshold value, which is proven a difficult problem when it comes to estimate it.
- A procedure of automatic estimation of the threshold is presented in the end of the second chapter, in the form of a new algorithm A-BIRCH that works the same way.
- Some concluding remarks related to the topic.

BIRCH ADVANTAGES

1. It is local, so each clustering decision is made without scanning all data points and currently existing clusters.

2. It exploits the observation that data space is not usually uniformly occupied and not every data point is equally important.

3. It makes full use of available memory to derive the finest possible sub-clusters while minimizing I/O costs.

4. It is also an incremental method that does not require the whole dataset in advance.

A-BIRCH

automatic threshold estimation

tree-BIRCH and cluster count

Gap Statistic

R_{max} and D_{min}

DANGERS

Handling Outliers	●●●●○
Natural Cluster Resemblance	●●○○○
Overall performances	●●●○○