Comparison of distributed evolutionary k-means clustering algorithms

M.C. Naldi, R.J.G.B. Campello

1. Introduction

Data clustering is a fundamental conceptual problem in data mining, in which one aims at determining a finite set of categories to describe a data set according to similarities among its objects. This problem has broad applicability in areas that range from image and market segmentation to document categorization, bioinformatics, and distributed computing, just to mention a few.

Many clustering algorithms have been proposed in the literature. Among them, the k-means method has been investigated for more than half a century. Recently, k-means has been elected one of the ten most influential data mining algorithms for being simple, scalable, and easily modifiable to a variety of contexts and application domains. However, exact distributed versions of k-means are still sensitive to the selection of the initial cluster prototypes and require the number of clusters to be specified in advance. Additionally, preserving data privacy among repositories may be a complicating factor. In order to overcome k-means limitations, two different approaches were adopted in this paper: the first obtains a final model identical to the centralized version of the clustering algorithm and the second generates and selects clusters for each distributed data subset and combines them afterwards. It is also described how to apply the algorithms compared while preserving data privacy. The algorithms are compared experimentally from two perspectives: the theoretical one, through asymptotic complexity analyses, and the experimental one, through a comparative evaluation of results obtained from a collection of experiments and statistical tests. The results obtained indicate which algorithm is more suitable for each application scenario.

© 2015 Elsevier B.V. All rights reserved.
Distributed Clustering (CDC) were proposed, which are a category of algorithms based on different validity indices have never been extensively compared. Comparisons among different algorithms have been suggested, though usually with less data transmission or computational load (rather than distributed) processing. An additional complicating factor resides in preserving data privacy, which is a legal obligation in some European countries and the United States, among other countries [25]. In some scenarios, the data may be analyzed inside the repository it belongs to, but cannot be shared with any other repositories. One such example is a collaboration among different companies to obtain an improved model identical to a hypothetical model generated by a centralized algorithm are usually applied to distributed subsets of the data and, later, the results are combined into a final solution [33]. An overview of a typical DDM application is illustrated in Fig. 1.

The DDM algorithms can be categorized into exact or approximate [34]. On the one hand, exact algorithms produce a final model identical to a hypothetical model generated by a centralized algorithm with access to the full data set. On the other hand, approximate algorithms produce a model that approximates a centralized model, usually with less data transmission or computational savings.

A review of DDM techniques can be found in [32] and an extensive DDM bibliography can be consulted in [35]. In order to meet the increasing need for distributed computational techniques with good performance and scalability, distributed versions of classic clustering algorithms have been proposed. One of the most cited distributed versions of the k-means algorithm was proposed by [36], later improved by [37] and adapted to peer-to-peer networks by [38,39]. Ref. [40] proposed a technique to parallelize algorithms based on centroids, which includes not only the k-means algorithm, but others like the Expectation Maximization [27] and BIRCH [26] algorithms. Other papers proposed the distribution of hierarchical clustering algorithms with the main objective of dividing the calculation of data dissimilarity among different processing units [41,42]. Ref. [28] proposed a parallel version of the BIRCH algorithm that balances the computational load among processors in a cyclic manner. Like the k-means, hierarchical algorithms were also adapted to peer-to-peer networks [34]. Ref. [43] developed a partitioning-distributed clustering algorithm that represents the whole data set. CDC algorithms were successfully applied to distributed data, specially for scenarios where the number of clusters is unknown. In particular, a novel comparison among DF-EAC and CDC algorithms using different validity indices was conducted based on two perspectives: the theoretical one, through asymptotic complexity analyses, and the experimental one, through a comparative evaluation of results obtained from a collection of experiments and statistical tests. Additionally, DF-EAC was revisited and two modifications were investigated: the first preserves data privacy among repositories and the second uses an alternative relative validation index to evaluate the resulting clusters. The modified DF-EAC variants are also compared in this study.

The remainder of this paper is organized as follows. In Section 2, a brief description of the area within which this study falls is provided. Then, in Section 3, the DF-EAC is presented, followed by a description of how it is distributed and of its complexity analysis. The CDC algorithms are described in Section 4. In Section 5, the DF-EAC and CDC algorithms are experimentally compared in order to determine which algorithms are most appropriate for each application scenario. Finally, the conclusions are addressed in Section 6.

2. Distributed clustering and privacy preservation

According to [32], DDM techniques involve discovering patterns or generating models from distributed data for which centralization is neither feasible nor desirable. In order to solve this problem, different algorithms or different parts of one algorithm are usually applied to distributed subsets of the data and, later, the results are combined into a final solution [33].

A review of DDM techniques can be found in [32] and an extensive DDM bibliography can be consulted in [35]. In order to meet the increasing need for distributed computational techniques with good performance and scalability, distributed versions of classic clustering algorithms have been proposed. One of the most cited distributed versions of the k-means algorithm was proposed by [36], later improved by [37] and adapted to peer-to-peer networks by [38,39]. Ref. [40] proposed a technique to parallelize algorithms based on centroids, which includes not only the k-means algorithm, but others like the Expectation Maximization [27] and BIRCH [26] algorithms. Other papers proposed the distribution of hierarchical clustering algorithms with the main objective of dividing the calculation of data dissimilarity among different processing units [41,42]. Ref. [28] proposed a parallel version of the BIRCH algorithm that balances the computational load among processors in a cyclic manner. Like the k-means, hierarchical algorithms were also adapted to peer-to-peer networks [34]. Ref. [43] developed a partitioning-distributed clustering algorithm that represents the whole data set. CDC algorithms were successfully applied to distributed data, specially for scenarios where the number of clusters is unknown. In particular, a novel comparison among DF-EAC and CDC algorithms using different validity indices was conducted based on two perspectives: the theoretical one, through asymptotic complexity analyses, and the experimental one, through a comparative evaluation of results obtained from a collection of experiments and statistical tests. Additionally, DF-EAC was revisited and two modifications were investigated: the first preserves data privacy among repositories and the second uses an alternative relative validation index to evaluate the resulting clusters. The modified DF-EAC variants are also compared in this study.

The remainder of this paper is organized as follows. In Section 2, a brief description of the area within which this study falls is provided. Then, in Section 3, the DF-EAC is presented, followed by a description of how it is distributed and of its complexity analysis. The CDC algorithms are described in Section 4. In Section 5, the DF-EAC and CDC algorithms are experimentally compared in order to determine which algorithms are most appropriate for each application scenario. Finally, the conclusions are addressed in Section 6.
دریافت فوری متن کامل مقاله

امکان دانلود نسخه تمام متن مقالات انگلیسی
امکان دانلود نسخه ترجمه شده مقالات
پذیرش سفارش ترجمه تخصصی
امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
امکان دانلود رایگان ۲ صفحه اول هر مقاله
امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
دانلود فوری مقاله پس از پرداخت آنلاین
پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات