Pattern Recognition 43 (2010) 2698–2711

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Quantization-based clustering algorithm

Zhiwen Yu [a,b], Hau-San Wong [b,*]

[a] School of Computer Science and Engineering, South China University of Technology, China
[b] Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 4 May 2009
Received in revised form
6 February 2010
Accepted 24 February 2010

Keywords:
Histogram
Clustering algorithm
K-means

ABSTRACT

In this paper, a quantization-based clustering algorithm (QBCA) is proposed to cluster a large number of data points efficiently. Unlike previous clustering algorithms, QBCA places more emphasis on the computation time of the algorithm. Specifically, QBCA first assigns the data points to a set of histogram bins by a quantization function. Then, it determines the initial centers of the clusters according to this point distribution. Finally, QBCA performs clustering at the histogram bin level, rather than the data point level. We also propose two approaches to improve the performance of QBCA further: (i) a shrinking process is performed on the histogram bins to reduce the number of distance computations and (ii) a hierarchical structure is constructed to perform efficient indexing on the histogram bins. Finally, we analyze the performance of QBCA theoretically and experimentally and show that the approach: (1) can be easily implemented, (2) identifies the clusters effectively and (3) outperforms most of the current state-of-the-art clustering approaches in terms of efficiency.

1. Introduction

As is well known, clustering is one of the classical problems in many different applications [1–9], such as pattern recognition, multimedia, data mining, knowledge discovery and data compression. The objective of clustering is to partition the data into different groups. Data which are in the same group have higher similarity, while data which originate from different groups have lower similarity. Based on various criteria, a large number of approaches for finding good clusters have been proposed in different applications [48], such as clustering based on hierarchical relationship [1–5], clustering based on density [6,7], clustering based on a grid structure [8,9], and clustering based on partition [10–30]. One of the most important techniques is K-means and its variants [10–30] in which a well-defined objective function is minimized. Suppose (i) the data set P consists of n data points ($P=\{\mathbf{p}_1,\ldots,\mathbf{p}_n\}$) and (ii) each data point $\mathbf{p}_i$ has m attributes ($\mathbf{p}_i=\{p_{i\,1},\ldots,p_{im}\}$), the objective function of K-means is to search for a set of cluster centers $S=\{\mathbf{s}_1^*,\mathbf{s}_2^*,\ldots,\mathbf{s}_k^*\}$ which minimize the sum of squared error:

$$\{\mathbf{s}_1^*,\mathbf{s}_2^*,\ldots,\mathbf{s}_k^*\}=\arg\min_{\{\mathbf{s}_1,\mathbf{s}_2,\ldots,\mathbf{s}_k\}\subset\mathbb{R}^m}\sum_{h=1}^{k}\sum_{\mathbf{p}_i\in P_h}\|\mathbf{s}_h-\mathbf{p}_i\|^2 \quad (1)$$

Compared with other clustering algorithms, the K-means family of algorithms has three major advantages: (i) simple implementation; (ii) efficient when handling a large data set; and (iii) a solid theoretical foundation based on the greedy optimization of the Voronoi partition. As a result, K-means is widely used in real world applications.

Some of these applications require the processing of a very large number of data points. For example, during the process of image segmentation, hundreds of thousands of pixels in an image are required to be separated into several groups [33-36]. Each pixel is viewed as a point consisting of multiple attributes, such as the RGB color values. As another example, with the pervasiveness of mobile devices and wide availability of wireless networks, a large number of location data are generated by the mobile users. Although traditional K-means and its variants can be applied to cluster these data, most of them are not very effective in handling this kind of large data set.

In this paper, we propose a quantization-based clustering algorithm (QBCA) for classifying a large number of data points, which processes large sets of data points in an efficient and effective way. Compared with traditional clustering algorithms, QBCA achieves low computation time without sacrificing the quality of the clusters. The approach is composed of three steps. (1) After assigning all the data points to a set of histogram bins by a quantization function, QBCA first selects the initial centers of the clusters according to the number of points assigned to each histogram bin. (2) Then, it makes use of specific properties of the minimum and maximum values of a distance measure to perform clustering at the histogram bin level. (3) Finally, QBCA assigns the points to their corresponding clusters. The major idea of QBCA is to perform clustering at the histogram bin level, rather than the data point level. Two approaches are proposed to improve the performance of QBCA further: (i) a shrinking process is applied to

* Corresponding author. Tel.: +852 27888624.
E-mail address: cshswong@cityu.edu.hk (H.-S. Wong).

0031-3203/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2010.02.020

the histogram bins to reduce the number of distance computations and (ii) a hierarchical structure is constructed to perform efficient indexing on the histogram bins. We compare QBCA with traditional clustering algorithms, which are variants of the K-means algorithm, on synthetic data sets and real data sets. The results of the experiments show that QBCA outperforms most of the current algorithms in terms of efficiency.

The contribution of the paper is twofold. First, we design a quantization-based clustering algorithm (QBCA) to classify a large number of data points. Second, we propose two approaches, namely the shrinking process and the hierarchical structure, to improve the performance of QBCA further.

The remainder of the paper is organized as follows. Section 2 describes previous related works on clustering algorithms. Section 3 provides the definitions of a number of new terms related to the proposed approach. Section 4 introduces the quantization-based clustering algorithm (QBCA). Section 5 describes the shrinking process for improving the performance of QBCA. Section 6 describes the hierarchical structure for histogram bin indexing. Section 7 evaluates our proposed algorithms experimentally, and Section 8 provides our conclusion and describes future research directions.

2. Related work

Although there are many sophisticated clustering algorithms which are recently proposed [1–9], such as CURE [1], CHAMELEON [2], ROCK [3], GRIN [4], BIRCH [5], DBSCAN [6], OPTICS [7], STING [8], CLIQUE [9], the most popular algorithms for performing clustering on the data set with a large number of data points remain those belonging to the K-means family [10–30]. Due to their efficiency and effectiveness to handle large sets of data points, the algorithms in the K-means family are very important for efficient clustering. Since our proposed algorithm belongs to the K-means family, we only consider these algorithms in our subsequent description.

K-means partitions the data points into k clusters to minimize an objective function $f(\mathbf{S},\mathbf{W})$.

$$f(\mathbf{S},\mathbf{W}) = \sum_{h=1}^{k}\sum_{i=1}^{n}\sum_{l=1}^{m} \omega_{i,h} \cdot d(s_{h,l},p_{i,l}) \tag{2}$$

subject to

$$\sum_{h=1}^{k} \omega_{i,h} = 1 \tag{3}$$

where $\mathbf{S}$ is a set of cluster centers ($\mathbf{S}=\{\mathbf{s}_1, \ldots, \mathbf{s}_k\}$), and $d(s_{h,l},p_{i,l})$ denotes the distance between the center $\mathbf{s}_h$ of the h th cluster and the data point $\mathbf{p}_i$ in the l-th attribute. $\mathbf{W}$ is an $n \times k$ partition matrix, and $\omega_{i,h}$ is its constituent variable: If $\omega_{i,h}=1$, the data point $\mathbf{p}_i$ belongs to the h th cluster.

If the attribute is numeric,

$$d(s_{h,l},p_{i,l}) = (s_{h,l}-p_{i,l})^2 \tag{4}$$

If the attribute is categorical,

$$d(s_{h,l},p_{i,l}) = \begin{cases} 1 & \text{if } p_{i,l} \neq s_{h,l} \\ 0 & \text{if } p_{i,l} = s_{h,l} \end{cases} \tag{5}$$

Clearly, if all attributes in the data are categorical, K-means is equivalent to its variant K-modes [20]. On the other hand, if the data contains both categorical values and numerical values, K-means becomes equivalent to its variant K-prototypes [21].

In particular, K-means can be considered as a special case of the Expectation–Maximization (EM) algorithm [31]. As a result, the process of K-means mainly consists of two steps. In the expectation step, K-means fixes the matrix $\mathbf{S}$, which is the set of cluster centers, and determines the matrix $\mathbf{W}$ as follows:

$$d(\mathbf{s}_{h^*},\mathbf{p}_i) = \min_{h=1\ldots k} d(\mathbf{s}_h,\mathbf{p}_i) \tag{6}$$

$$\omega_{i,h} = \begin{cases} 1 & \text{if } h = h^* \ \ \forall h \in \{1,\ldots, \ \ k\} \\ 0 & \text{if } h \neq h^* \end{cases} \tag{7}$$

In the maximization step, K-means fixes the matrix $\mathbf{W}$, and determines the matrix $\mathbf{S}$ as follows:

$$\mathbf{s}_h = \frac{\sum_{i=0}^{n_h} \omega_{i,h}\mathbf{p}_i}{\sum_{i=0}^{n_h} \omega_{i,h}} \quad \forall h \in \{1,\ldots,k\} \tag{8}$$

where n_h denotes the number of data points which are assigned to the h th cluster.

Recently, there are a lot of research works which focus on how to improve the performance of K-means. For example, Laszlo et al. [41] proposed to adopt a genetic algorithm (GA) to search for the centers of K-means with the help of a hyper-quadtree constructed on the data, in order to obtain the global optimal value of the objective function in K-means. They also designed a novel crossover operator that exchanges neighboring centers during the genetic optimization process [42], and selected cluster centers as the initial seeds for K-means. Bandyopadhyay et al. [43] designed a new point symmetry-based distance measure to improve the performance of K-means. Chung et al. [44] proposed the modified point symmetry-based K-means (MPSK) algorithm which works well for both the symmetrical intra-clusters and the symmetrical inter-clusters. Lai et al. [45] partitioned cluster centers into inactive and active sets based on the information of cluster displacement. They reduced the computational complexity of K-means by only focusing on cluster centers in the active set. Lai et al. [46] further improved the efficiency of K-means by the displacement of cluster centers to reject useless candidates for a data point. Chang et al. [47] improved the genetic algorithm based K-means by adopting adaptive probabilities for the crossover and mutation operators.

Although there are different variants of K-means [20–30], one of the most efficient algorithms in the K-means family is the filtering algorithm which is based on the kd-tree [10]. The filtering algorithm (i) stores the data points in a kd-tree and (ii) maintains a subset of candidate centers for each node of the kd-tree. When the candidates are propagated to the node's children, some of them are pruned. The lower the level the node is at, the smaller the number of candidates the node contains. During the K-means clustering process, we do not need to consider all the k centers, but only those candidate centers in the node which are associated with the current data point. In other words, the filtering algorithm achieves low computation time by reducing the distance computation cost. Based on the filtering algorithm, different K-means variants are proposed, which include: (i) the swap algorithm (SWAP), which applies perturbation to the current set of cluster centers to escape from local minima. Specifically, selected centers are swapped between the current center set and a reserved list of candidate centers; (ii) the simple hybrid algorithm (EZ-Hybrid), which performs a single swap followed by a pre-specified number of K-means iterations; and (iii) the Hybrid algorithm (Hybrid), which performs a selected number of swaps followed by multiple K-means iterations. However, the algorithm needs $O(n \log n)$ time complexity to construct a kd-tree, and the resulting computation time cannot satisfy the requirement of efficient clustering, especially when the set of pixels is large. The motivation of this paper is to reduce the clustering time further by substituting the kd-tree with a histogram-based indexing structure, since the time complexity of constructing a histogram by a suitable quantization function is $O(n)$.