Knowledge-Based Systems 49 (2013) 123–133

Contents lists available at SciVerse ScienceDirect

Knowledge-Based Systems

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

A sample-based hierarchical adaptive *K*-means clustering method for large-scale video retrieval

Kaiyang Liao, Guizhong Liu *, Li Xiao, Chaoteng Liu

School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China

ARTICLE INFO

Article history:
Received 22 July 2012
Received in revised form 1 May 2013
Accepted 7 May 2013
Available online 20 May 2013

Keywords:
Data clustering
K-means algorithm
Pattern recognition
Content-based video copy detection
Large-scale data

ABSTRACT

Finding useful patterns in large datasets has attracted considerable interest recently, and one of the most widely studied problems in this area is the identification of clusters in a multi-dimensional dataset. This paper introduces a sample-based hierarchical adaptive *K*-means (SHAKM) clustering algorithm for large-scale video retrieval. To handle large databases efficiently, SHAKM employs a multilevel random sampling strategy. Furthermore, SHAKM utilises the adaptive *K*-means clustering algorithm to determine the correct number of clusters and to construct an unbalanced cluster tree. Furthermore, SHAKM uses the fast labelling scheme to assign each pattern in the dataset to the closest cluster. To evaluate the proposed method, several datasets are used to illustrate its effectiveness. The results show that SHAKM is fast and effective on very large datasets. Furthermore, the results demonstrate that the proposed method can be used efficiently and successfully for a project on content-based video copy detection.

1. Introduction

Cluster analysis plays an indispensable role in exploring the underlying structure of a given dataset, and it is widely used in a variety of engineering and scientific subjects, such as biology, medicine, sociology, psychology, pattern recognition, and image processing and retrieval. The primary objective of cluster analysis is to partition a given set of patterns (which are usually represented as a vector of measurements or a point in a multidimensional space) into so-called homogeneous clusters based on their similarity. Intuitively, the patterns within a valid cluster are more similar to each other than the patterns that belong to different clusters.

Cluster analysis is very experiment-oriented, and a cluster algorithm that can efficiently address all situations is not yet available. The concept of what constitutes a good cluster depends on the application, and there are many methods for validating the clustering results using various criteria. Extensive and excellent overviews of clustering algorithms can be found in the literature [6,18,19,21,35,37]. Perhaps the most widely used and studied member of the clustering family is the *K*-means algorithm [3]. Recently, neural networks, including self-organising feature maps [27,32], competitive-learning networks [24], and adaptive resonance theory networks [7,8], have also been used to cluster data. Spectral clustering algorithms [11,36], which exploit pairwise similarities of data samples using eigen decomposition of their similarity matrix, have been shown to be successful in the area of data mining. Additionally, Chakrabarti et al. [9] proposed an evolutionary hierarchical clustering algorithm and an evolutionary *K*-means clustering algorithm.

There are many applications in which it is necessary to cluster a large number of points. The meaning of 'large' has varied (and will continue to do so) with the development of science and technology (for example, memory and CPU). In the 1960s, 'large' meant several thousand patterns. Starting in the 2000s, clustering algorithms have had to address millions of patterns. Currently, there are applications in which several billion patterns with high dimensionality must be clustered. For example, in image/video retrieval, billions of patterns with a dimensionality of more than one hundred must be clustered to achieve data abstraction. The number of clusters is becoming larger and larger for many applications, such as video retrieval and image classification. In many approaches, the algorithm complexity is proportional to the number of clusters. The costs of addressing large datasets are always important, and the struggle between the computational time and the cluster numbers becomes severe, especially as the cluster numbers increase. It is desirable to account for the amount of time that a user is willing to wait for the results of the clustering algorithm [39].

For the most part, the clustering algorithms presented above cannot be directly applied to very large datasets because of their time complexity with respect to the input size. To overcome these urgent problems, an improved cluster scheme based on hierarchical *K*-means (SHAKM) is proposed. To handle large databases, SHAKM employs a multilevel random sampling strategy to address it efficiently. Furthermore, SHAKM utilises the adaptive *K*-means clustering algorithm to determine the correct number of clusters

* Corresponding author. Tel./fax: +86 29 82667836.
E-mail address: liugz@mail.xjtu.edu.cn (G. Liu).

0950-7051/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.knosys.2013.05.003

and to construct an unbalanced cluster tree. Furthermore, SHAKM uses the fast labelling scheme to assign each pattern in the dataset to the closest cluster. The features of the proposed SHAKM clustering algorithm are the following: (1) The clustering algorithm is especially suitable for very large databases. (2) The algorithm can be used to generate a large number of clusters. (3) The algorithm has very low time complexity and space complexity.

The remainder of this paper is organised as follows: Section 2 reviews the related work. Section 3 presents our proposed SHAKM clustering algorithm. Section 4 introduces the evaluation criteria and the datasets for the experiments. Section 5 presents the experimental results. Finally, the paper is concluded in Section 6.

2. Related work

We are given a set of input patterns $X = \{x_1, \ldots, x_i, \ldots, x_n\}$, where $x_i = (x_{i1}, x_{i2}, \ldots, x_{id})^T \in R^d$, and we are given that each component x_{ij} is an attribute (feature, variable, or dimension).

The hard partitional clustering algorithms attempt to seek a K-partition of X, $C = \{C_1, \ldots, C_k\} (k \leqslant n)$ that satisfies the following conditions:

$$C_i \neq \phi, \quad i = 1, \ldots, k \tag{1}$$

$$\cup_{i=1}^{k} C_i = X \tag{2}$$

$$C_i \cap C_j = \phi, \quad i,j = 1, \ldots, k \quad and \quad i \neq j \tag{3}$$

The most intuitive and frequently used criterion function in partitional clustering techniques is the squared error criterion, which tends to work very well with isolated and compact clusters. The squared error for a clustering C of a pattern set X (which contains k clusters) is

$$e^2(X,C) = \sum_{j=1}^{k} \sum_{i=1}^{n_j} \left\| X_i^j - c^j \right\|^2 \tag{4}$$

where X_i^j is the ith pattern that belongs to the jth cluster and c^j is the centroid of the jth cluster.

The K-means algorithm is the best-known and the most widely used squared error-based clustering algorithm [14,25]. The K-means clustering algorithm has the following description:

(1) Choose k cluster centres to coincide with k randomly chosen patterns or based on some prior knowledge. Calculate the cluster prototype matrix M.

$$M = [m_1, \ldots, m_k] \tag{5}$$

where m_i is the sample mean for the ith cluster.

(2) Assign each pattern in the dataset to the closest cluster C_l, i.e.,

$$\begin{aligned} & x_j \in C_l, \quad if \quad \|x_j - m_l\| < \|x_j - m_i\| \\ & \quad for \quad j = 1, \ldots, n, \quad i \neq l, \quad and \quad i,l = 1, \ldots k. \end{aligned} \tag{6}$$

(3) Recompute the cluster prototype matrix using the current cluster memberships.

(4) If a convergence criterion is not met, then go to step 2.

Usually, the convergence criteria are as follows: no reassignment of new patterns to the cluster centres or a minimal decrease in the squared error. The K-means algorithm is very simple and easy to implement in many practical applications. It can work very well for compact and hyperspherical clusters. The reasons for the popularity of the K-means algorithm include the following:

(1) The time complexity is $O(nkt)$, where n is the number of patterns in the dataset, k is the number of clusters, and t is the number of iterations taken by the algorithm to converge. In general, k and t can be fixed in advance; thus, the algorithm has linear time complexity in the size of the dataset [12].

(2) The space complexity is $O(k + n)$. It requires additional space to store the centroids.

(3) The algorithm is order independent. For a given initial seed set of cluster centres, it generates the same partition of the data irrespective of the order in which the patterns are presented to the algorithm [20].

Hierarchical clustering (HC) organises data into a hierarchical structure according to the proximity matrix, and it attempts to construct a tree-like nested structure partition of X

$$H = \{H_1, \ldots, H_v\} \quad v \leqslant k \tag{7}$$

such that $C_i \in H_m$, $C_j \in H_l$, $m > l$, and $m, l = 1, \ldots, v$, implying $C_i \in C_j$ or $C_i \cap C_j = \emptyset$. The root node of the tree represents the whole dataset, and each leaf node is regarded as a data item. The intermediate nodes represent the centres at all levels.

With further advances in database and internet technologies, clustering algorithms will face more and more severe challenges in handling the rapid growth of data. Various clustering approaches for large datasets have been introduced in the literature. These clustering approaches can be divided into four classes:

(1) Random sampling approach, for example, CLARA (clustering large applications) [23] and CURE [15]. CLARA represents each cluster with a centroid, whereas CURE uses a set of well-scattered and centre-shrunk points to represent each cluster. Bezdek et al. [4] introduced an approximate clustering in large relational data using progressive sampling. Additionally, Wang et al. [34] proposed an approximate pairwise clustering for large datasets via sampling plus extension.

(2) Divide-and-conquer approach. The best-known graph-theoretic divisive clustering algorithm constructs the minimal spanning tree (MST) of the data [38] and then deletes the MST edges with the largest lengths to generate the clusters. Bordogna and Pasi proposed an adaptive hierarchical fuzzy clustering algorithm that is a quality-driven divisive algorithm [5].

(3) Incremental approach. The BIRCH algorithm proposed by Zhang et al. [39] is a very good incremental clustering algorithm. Bagirov et al. proposed a fast modified global K-means algorithm for incremental cluster construction, which introduces an auxiliary cluster function to generate a set of starting points that lie in different parts of the dataset [2].

(4) Parallel implementation approach. Pizzuti and Talia proposed a parallel realisation Bayesian approach that is used in auto classification to find out the optimal partition of the given dataset based on the prior probabilities [30].

Unlike in existing studies, the technique proposed in this paper employs multilevel random sampling to handle large databases and utilises the adaptive K-means clustering algorithm to determine the correct number of clusters.

3. Sample-based hierarchical adaptive K-means algorithm

3.1. Multilevel random sampling

If a small amount of a sample dataset can represent the entire dataset well, then we will be able to obtain information similar to the global clusters from cluster-analysing the sample dataset.