Information Sciences 181 (2011) 4658–4672

Contents lists available at ScienceDirect

Information Sciences

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

A novel ant-based clustering algorithm using the kernel method

Lei Zhang *, Qixin Cao

Research Institute of Robotics, State Key Lab of Mechanical System and Vibration, Shanghai Jiao Tong University, No. 800 Dongchuan Road, Shanghai 200240, China

ARTICLE INFO

Article history:
Available online 12 November 2010

Keywords:
Ant-based clustering
Kernel
Swarm intelligence
Kernel principal component analysis

ABSTRACT

A novel ant-based clustering algorithm integrated with the kernel (ACK) method is proposed. There are two aspects to the integration. First, kernel principal component analysis (KPCA) is applied to modify the random projection of objects when the algorithm is run initially. This projection can create rough clusters and improve the algorithm's efficiency. Second, ant-based clustering is performed in the feature space rather than in the input space. The distance between the objects in the feature space, which is calculated by the kernel function of the object vectors in the input space, is applied as a similarity measure. The algorithm uses an ant movement model in which each object is viewed as an ant. The ant determines its movement according to the fitness of its local neighbourhood. The proposed algorithm incorporates the merits of kernel-based clustering into ant-based clustering. Comparisons with other classic algorithms using several synthetic and real datasets demonstrate that ACK method exhibits high performance in terms of efficiency and clustering quality.

1. Introduction

Clustering is a method that divides a dataset into groups of similar objects, thereby minimizing the similarities between different clusters and maximizing the similarities between objects in the same cluster. Clustering is widely applied in data mining, such as in document clustering and Web analysis. Classic clustering approaches include partition-based methods, such as *K*-means, *K*-medoids, and *K*-prototypes [20,22]; hierarchy-based methods, such as BIRCH [44]; density-based methods [1,10]; grid-based methods [43]; and model-based methods, such as neural networks and self-organizing map (SOM) [4,30].

Recently, ant-based clustering, which is a type of clustering algorithm that imitates the behaviour of ants, has earned researchers' attention. Ant-based clustering can be divided into two classes. The first class imitates the ant's foraging behaviour, which involves finding the shortest route between a food source and the nest. This intelligent behaviour is achieved by means of pheromone trails and information exchange between ants. Shelokar et al. [34] and Chen et al.'s [5] proposed algorithms belong to this class. These algorithms treat clustering as an optimization task and utilize ant colony optimization (ACO) methods to obtain optimal clusters. An extension of ACO, called constrained ACO (C-ACO) [6], was suggested to cluster data involving arbitrary shapes or outliers. A variant of ACO, called the aggregation pheromone density-based clustering algorithm (APC), was also suggested [12,13]. Similar to ACO, APC is based on the aggregation pheromones found in ants. The advantage of these methods is that the objective function is explicit. The key elements of these algorithms are the pheromone matrix updating rule and the heuristic function.

The second class imitates ants' behaviour of clustering their corpses and forming cemeteries. Some ants can pick up dead bodies randomly distributed in the nest and group them into different sizes. The large group of bodies attracts the ants to

* Corresponding author.
E-mail addresses: zhanglei75@sina.com, zhanglei@sjtu.edu.cn (L. Zhang).

0020-0255/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2010.11.005

deposit more dead bodies and becomes larger and larger. The essence of this phenomenon is positive feedback [25]. One of the first studies related to this domain is the work of Deneubourg [9], who came up with the basic model (BM) to explain the ants' movement. In the BM, the ants move randomly and pick up or drop objects according to the number of similar surrounding objects to cluster them. Lumer and Faieta [24] extended the model and applied it to data analysis (they called this the LF algorithm). In their analysis, an object with n attributes can be viewed as a point in the R^n space. The point is projected into a low-dimensional space (often a two-dimensional plane). The similarity of the object with those in the local neighbourhood is calculated to determine whether the object should be picked up or dropped by ants. As a basic algorithm, LF was followed and improved by a number of modified algorithms in different applications. Wu et al. [37,38] further explained the idea of the similarity coefficient and suggested a more simple probability conversion function. Ramos and Merelo [29] studied ant-based clustering with different ant speeds to cluster text documents. Yan et al. [40,41] suggested multiple ant colonies consisting of independent colonies and a queen ant agent. Each ant colony had a different moving speed and probability conversion function. The hypergraph model was used to combine the results of all parallel ant colonies.

In addition to the above-mentioned studies, a series of research by Handl deserves special attention. She came up with a set of strategies for increasing the robustness of the LF algorithm and applying it to document retrieval [15]. She performed a comparative study of ant-based clustering with *K*-means, average links, and 1d-SOM [16,17]. An improved version, ATTA, which incorporates adaptive and heterogeneous ants and time-dependent transporting activity, was proposed in her latest paper [18]. The main feature of this kind of algorithm is that the algorithm directly imitates the ant's behaviour to cluster data and the clustering objective is implicitly defined [19].

Beyond the two classes of ant-based clustering, Tsang and Kwong [36] proposed ant colony clustering for anomaly intrusion detection. This method integrates the characteristics of the two above-mentioned classes. Specifically, cluster formation and searching for an object are regarded as nest building and food foraging, respectively. The ants exhibit picking up and dropping behaviours while simultaneously depositing cluster-pheromones on the grid. Xu et al. [39] suggested a novel ant movement model wherein each object was viewed as an ant. The ant determines its behaviour according to the fitness of its local neighbourhood. Essentially, this model is similar to that in the second class of ant-based clustering.

Combinations of ant-based clustering with other clustering methods can also be found. For example, ant-based clustering has been combined with *K*-means [26] and with *K*-harmonic means [21]; ant colonies have been hybridized with fuzzy *C*-means [28,42]; fuzzy ants have been endowed with intelligence in the form of IF–THEN rules [23]; and the hybrid approach has been generated based on particle swarm optimization (PSO), ACO, and *K*-means [27]. In these methods, the role of ant-based clustering is mainly to create initial clusters for other clustering algorithms.

A general simulation of swarm and collective intelligence has been described [33] as well as a comprehensive overview of ant-based and swarm-based clustering [19].

Our particular interest is in the second kind of ant-based clustering discussed above. Although ant-based clustering has been modified gradually, it still needs improvement in terms of its applications. The focus of our work is on the following two important problems:

- Improving the algorithm's efficiency
  Ant-based clustering can be implemented through the parallel computing of each ant [7], which may lead to the development of an efficient algorithm. However, it is not highly efficient because of the randomness in the algorithm. Initially, the objects are randomly projected onto the toroidal grid; thus, the similarities of the objects in a local neighbourhood are very low. Therefore, the objects are easily picked up but not easily dropped by the ants. It takes long time to go from the inception of the algorithm to the moment when the rough clusters are created. Commonly, 100,000 iterations are needed for ant-based clustering algorithms [2].
- Improving the algorithm's clustering quality
  In essence, ant-based clustering algorithms are distance-based because the similarity of the objects is computed by Euclidean distance or Cosine distance. Just like other distance-based clustering algorithms, they are effective for datasets with an ellipsoidal or Gaussian distribution. If the separation boundaries between clusters are nonlinear, however, the algorithms will fail. An alternative approach to solving this problem is kernel mapping, which transforms the data into a high-dimensional feature space and then performs the clustering in the feature space.

Kernel-based clustering was proposed by Mark [14]. Its integration with *K*-means, fuzzy *K*-means, SOM, and support vector machines has been shown to be effective in improving clustering quality [11].

In this paper, we incorporated the kernel method into ant-based clustering and created the novel ant-based clustering with the kernel method (ACK). The applications of kernels in ACK are shown in two respects. First, kernel principal component analysis (KPCA) is used to modify the initial projection of all objects. Second, the Euclidean distance in the feature space is applied as a measure of the similarity between the objects. These two applications are geared toward solving the problems mentioned above. Compared with general ant-based clustering, ACK can create better clustering results in some datasets with non-Gaussian distribution. Moreover, the clustering quality and efficiency are greatly improved.

The paper is organized as follows: Section 2 describes the basics of the ant-based clustering algorithm. Section 3 introduces kernel-based clustering. Section 4 proposes the novel ant-based clustering algorithm using the kernel method. Section 5 compares the proposed algorithm with other clustering algorithms. Finally, Section 6 gives the conclusions and discusses future work.