A New Ant-based Clustering Algorithm on High Dimensional Data Space

CHEN Jianbin[a,1], Sun Jie[b],CHEN Yunfei[c]

[a]Associate professor，Business College of Beijing Union University, Beijing, CN
[b,c] Business College of Beijing Union University, Beijing, CN

Abstract. Ant-based clustering due to its flexibility, stigmergic and self-organization has been applied in a variety areas from problems arising in commerce, to circuit design, and to text-mining, etc. A new ant-based clustering method named AMC algorithm has been presented in this paper. Firstly, an artificial ant movement(AM) model is presented; secondly, the new ant clustering algorithm has been constructed based on AM model. In this algorithm, each ant is treated as an agent to represent a data object, each ant has two states: resting state and moving state. The ant's state is controlled by two predefined functions. By moving dynamically, the ants form different subgroups adaptively, and consequently the whole ant group dynamically self-organized into distinctive and independent subgroups within which highly similar ants are closely connected. This algorithm can be accelerated by the use of a global memory bank, increasing radius of perception and density-based 'look ahead' method for each agent. Experimental results show that the AMC algorithm is much superior to other ant clustering methods. It is adaptive, robust and efficient, and achieves high autonomy, simplicity and efficiency. It is suitable for solving high dimensional and complicated clustering problems.

Keywords. Ant-based heuristic, clustering, high-dimensional data space.

1. Introduction

Clustering analysis is an important method in data mining. It is a discovery process that groups a set of data such that the intra-cluster similarity is maximized and the inter[1]-cluster similarity is minimized. Clustering of data in a large dimension space is of a great interest in many data mining applications[10].

Clustering has been widely studied since the early 60's. Some classic approaches include hierarchical algorithms, partitioning method such as K-means, Fuzzy C-means, graph theoretic clustering, neural networks clustering, and statistical mechanics based techniques. Recently, several papers have highlighted

[1] Associate Professor, Business College of Beijing Union University,A3 Yanjingdongli, Chaoyang,Beijing,100025,P.R.CHINA; Phone : 086-10-65940712; Fax : 086-10-65940655;E-mail: jianbin.chen@bcbuu.edu.cn

the efficiency of stochastic approaches based on ant colonies for data clustering [2,3, 5,8].

Ant-based clustering and sorting was originally introduced for tasks in robotics by Deneubourg [3]. The entomologists who observe societies of ants found that larvae and food are not scattered randomly about the nest, but in fact they are sorted into homogenous piles. Deneubourg et al. proposed a basic model that explains the spatial structure of cemetery forms as a result of simple, local interactions without any centralized control or global representation of the environment[3]. Holland et al. applied related model to robotics to accomplish complex tasks by several simple robots[9]. Lumer and Faieta modified the algorithm to extend to numerical data analysis by introducing a measure of dissimilarity between data objects[5].Kuntz et al. applied it to graph-partitioning[11], text-mining[7] and VLSI circuit design[12]. Note that Monmarché has introduced an interesting AntClass algorithm, a hybridization of an ant colony with the k-means algorithm, and compared it to traditional k-means on various data sets, using the classification error for evaluation purposes[8]. However, the results obtained with this method are not applicable to ordinary ant-based clustering since it differs significantly from the latter.

The research presented here proposes a new Ant-based Clustering algorithm(AMC) which is enlightened by the behaviors of gregarious ant colonies. Firstly, an artificial ant movement(AM) model is presented; secondly, the new ant clustering algorithm has been constructed based on AM model. In this algorithm, each ant is treated as an agent to represent a data object, each ant has two states: resting state and moving state. The ant's state is controlled by two predefined functions. By moving dynamically, the ants form different subgroups adaptively, and consequently the whole ant group dynamically self-organized into distinctive and independent subgroups within which highly similar ants are closely connected. This algorithm can be accelerated by the use of a global memory bank, increasing radius of perception and density-based 'look ahead' method for each agent.

2. Clustering algorithms based on ant colony

Ant-based clustering was inspired by the clustering activities of corpses observed in real ant colonies[4]. The algorithm's basic principles are straightforward: Ants are modeled by simple agents that randomly move in their environment, a square grid with periodic boundary conditions. Data items that are scattered within this environment can be picked up, transported and dropped by the agents. The basic model proposed by Deneubourg et al and its extended clustering algorithms have three fatal problems: firstly, there are so many useless random movements for the agents before they picking or droping the data items; secondly, the parameter value of these algorithms are sensitive to be set initially, and result the clustering algorithm has no robusticity; thirdly, there are two kinds data to be stored and managed through the whole running time, one is ant agents and another is data items to be clustered. That's meaning that it needs more storage space and more complexity methods. Although Handl et al have made some effort to improve the

clustering performance[6], the clustering effect should be reformed through another side.

2.1 The Ant Movement Model（AM model）

Enlightened by the behaviors of gregarious ant colonies, an ant movement model has been proposed. The AM model simulates the demeanor of gregarious ant of seeking a comfortable environment to stay, defined the ant agent and two states for them, resting and moving. In the AM model, an artificial ant means a simple agent. It has a simple behavior that, when it has no comfortable site to be rest, it always moving; if it has find a appropriate site, it will stay there until it fills not comfortable again. Such and such, until all these agents find their site and stay there. Not similar with former methods that the ant agent want to carry data object to destination site, the artificial ant has been treated as an agent to represent a data object in our AM model. Thus, the moving of ant agent means the moving of data object.

In the AM model, the environment for ant agent to move about is mapped to a toroidal grid. In a toroidal grid, all the site have same neighborhood, have no difference of center, corner or border. We can defined a state variable q_i to describe the state of an ant agent o_i : $q_i=(x_i,y_i,c_i,s_i)(1\le i\le n)$, which n is the number of data object, (x_i,y_i) is the coordinate of the agent's site, c_i is the cluster ID and s_i is the state sign for resting or moving. The neighborhood can be defined as *Moore* neighborhood in which there are 8 neighbors for each site. For an ant agent, to determine resting or moving at time t , needs to calculate the fitness function $f(i)$ which also can be called similarity function.

2.2 Similarity and Distance

Let us assume that an ant o_i is located at site r at time t. The local density of objects similar to o_i at the site r is given by

$$f(i)=\begin{cases}\dfrac{1}{\sigma^2}\displaystyle\sum_{o_j\in Neigh_{\sigma\times\sigma}(r)}\left[1-\frac{d(o_i,o_j)}{\alpha}\right] & f(i)>0\\ 0 & \text{otherwise}\end{cases} \qquad (1)$$

Where, $f(i)$ is a modified version of Lumer et al.'s[5] neighborhood function. It is a measure of the average similarity of ant o_i with the other ant o_j present in its neighborhood. And $Neigh_{\infty}(r)$ is the local region. It is usually a square of $\sigma\times\sigma$ sites surrounding site r. $d(o_i,o_j)\in[0,1]$ is the distance of data object O_i with O_j in the space of attributes. It is usually Euclidean distance or Cosine distance. The parameter α is defined as similarity coefficient. It is a key coefficient that directly affects the number of cluster and convergence of the algorithm. Large values of α will result in making the similarity between the objects larger and forcing objects to lay the same clusters. When α is small, the similarity will decrease and may in the extreme result in too many separate clusters.

The parameter α also determines the cluster number and the speed of convergence. The bigger α is, the smaller the cluster number is, and the faster the algorithm converges.

2.3 Probability Activation Function

Probability activation function is the function which active an ant agent from a state to another state, resting or moving. Obviously similarity is one of its variables. The value domain is [0, 1]. There are two situation for an ant agent to exchange its state, and we defined two function for them to determine whether to exchange. The probability of an ant agent resting to moving will be calculated by $p_M(i)$:

$$p_M(i) = \left(\frac{k^+}{k^+ + f(i)} \right)^2 \qquad (2)$$

and the probability of an ant agent moving to resting will be calculated by $p_R(i)$:

$$p_R(i) = \left(\frac{f(i)}{k^- + f(i)} \right)^2 \qquad (3)$$

Where commonly $k^+ = 0.1$ and $k^- = 0.3$.

2.4 Basic Clustering algorithms based AM model

Based on the AM model, a basic generic algorithm is described in Algorithm 1.

Algorithm 1. Basic ant algorithm

```
1: procedure BASIC_ALGORITHM
 /*INITIALIZATION PHASE*/
2: data preprocess and initialize parameter
3: for each Agent do
4:    Randomly scatter Agent on the toridal grid
5:    let c_i =data item ID
6:    let s_i=moving
7:  end for
8: while(not termination)
9:  for each Agent do
10:    if s_i=moving  calculate p_R(i)
11:   if s_i=resting  calculate p_M(i)
12:   turn state based on p_R(i) or p_M(i)
13:  update c_i
14:  if s_i=moving select next site
 15: end for
16: end while
17: output cluster information of all Agents
```

3 Algorithm Optimization

In data clustering, many algorithms (like K-means and ISODATA [1]) require that an initial partition be given as input, before the data can be processed. This is the one common drawback of these methods, for it is not easy to specify a proper number of clusters for a set of data in advance. Moreover, these methods are often led to locally optimal by using a deterministic search, which is another major drawback of these methods.

Contrasting with those methods mentioned above, ant-clustering boasts a number of advantages due to the use of mobile agents, which are autonomous entities, both proactive and reactive, and have the capability to adapt, cooperate and move intelligently from one location to the other in the bi-dimensional grid space. These advantages are：

***Autonomy*:** Not any prior knowledge (like initial partition or number of classes) about the future classification of the data set is required. Clusters are formed naturally through ant's collective actions.

Flexibility: Rather than deterministic search, a stochastic one is used to avoid locally optimal.

***Parallelism*:** Agent operations are inherently parallel.

While many of those advantages look prefect, two important defects remain. The first one is that there may be some data objects which have never been assigned to ant when the algorithm is terminated. This is due to the fact that each time when an ant is assigned a new data object to inspect, the selected data object is randomly selected. Un-assigned objects lead to a high misclassification error rate of the algorithm, for they have never taken part into the clustering loop. The second defect is the long time consumption of clustering. Due to the random motion of the ants, ant-based clustering algorithms have a slow convergence rate.

To improve the performance of traditional ant-clustering algorithm, we have made some effort in it. We treat each data object as an ant agent to be sure that every object should be visited in the iteration. This method also can save memory because there are no additional agents beside data object themselves. Otherwise, we also adopt technologies such as memory bank, "look ahead" methods and so on.

4 Experiment results and analysis

We have applied the new algorithm AMC to several numerical databases including synthetic ones and real databases from the Machine Learning repository（Machine Learning Repository， http://www.ics.uci.edu/~mlearn/MLRepository.html ）. Synthetic database include: ANT1(75,2,4)、ANT2（500， 2， 4）、ANT3（800， 2， 9）, while real database are Iris（150， 4， 3） and Soybean(307,35,19).

We have used 4 evaluation measures to evaluate the resulting partition obtained by the three clustering algorithms. They are the number of identified clusters(#Clusters)、Inner Cluster Variance(Variance)，Classification Error Rate (Cl.Err) [8] and the overall running time of the algorithm(Runtime).

The results demonstrate that, if clear cluster structures exist within the data, the ant clustering algorithm including: CSI and AMC, is quite reliable at identifying the correct number of clusters. In contrast with the *k*-means, the AMC algorithm shows its strength in its ability to automatically determine the number of clusters within the data.

Compare the runtimes of the three algorithms, we can see AMC is the fastest algorithms and its time consumer changes little with the scale of data set. So it is a fast clustering algorithm with prefect scalability. The CSI algorithm is the slowest one of the three algorithms, but compared with the k-means, the increasing gradient of its time consumption decreases with the growth of data set.

5. Conclusion

In this paper, we have proposed a new ant-based clustering algorithm, which is derived from the AntClass, LF clustering algorithm and CSI. Firstly, the AM model has been proposed and a new clustering methods has been presented based on the AM model. Secondly, the device of memory bank is proposed, which can bring forth heuristic knowledge guiding ant moving in the bi-dimensional grid space. So the classification error rate drops subsequently. Thirdly, we proposed a density-based method permits each ant to “look ahead”, which reduces the times of region-inquiry. Consequently the clustering time gets saved. We made some experiments on real data sets and synthetic data sets. The experiments’ results are compared with those obtained using other classical clustering algorithm. The results demonstrated that AMC is a viable and effective clustering algorithm. Future work focuses in:

(1) How to give ant more powerful heuristic rule, which can guide the ant motion, therefore speed up the clustering rate.

(2) Combines AMC with the other clustering algorithm such as k-means and DBSCAN to further improve the clustering quality.

Reference：

[1] Ball G.H and Hall D.J.,ISODATA, a novel method of data analysis and pattern classification, Technical report, Stanford Research Institute,1965

[2] B.wu,Y.zheng,S.liu and Z.shi, SIM:A Document Clustering Algorithm Based on Swarm Intelligence. IEEE World Congress on Computational Intelligence,Hawaiian,PP.477-482.2002

[3] Deneubourg J L , Goss S , Frank N , Sendova-hanks A ,Detrain C ,Chrerien L. The dynamics of collective sorting : robot-like ants and ant-like robots. In : Proceedings of the 1st International Conference on Simulation of Adaptive Behavior : From Animals to Animats , MIT Press/ Bradford Books , Cambridge , MA , 1991. 356～363

[4] E.Bonabeau, M.Dorigo, and G. Theraulaz. Swarm Inteligence—From Natural to Artificial System. Oxford University Press, New York, NY,1999.

[5] E. Lumer and B. Faieta. Diversity and adaption in populations of clustering ants. In Proceedings of the Third International Conference on Simulation of Adaptive

Behaviour: From Animals to Animats 3, pages . 501–508. MIT Press, Cambridge, MA, 1994.

[6] Handl J, Meyer B. Improved ant-based clustering and sorting in a document retrieval interface. LNC 2439,2002.913-923.

[7] K. Hoe, W. Lai, and T. Tai. Homogenous ants for web document similarity modeling and categorization. In Proceedings of the Third International Workshop on Ant Algorithms (ANTS 2002), volume 2463 of *LNCS*, pages 256–261. Springer-Verlag, Berlin, Germany, 2002.

[8] Nicolas Monmarché, Mohamed Slimane, Gilles Venturini. AntClass: discovery of clusters in numeric data by an hybridization of an ant colony with the Kmeans algorithm，Internal Repport No 213,E3i,January 1999

[9] O.E.Holland and C.Melhuish. Stigmergy, self-organization, and sorting in collective robotics, Artificial Life,5,1999,pp.173-202

[10] P.Berkhin. Survey of Clustering Data Mining Techniques. Accrue Software Research Paper.2002.

[11] P.Kuntz, D. Snyers, and P. Layzell. A stochastic heuristic for visualizing graph clusters in a bi-dimensional space prior to partitioning. Journal of Heuristics, 5(3):327–351, 1998.

[12] P.Kuntz,P.Layzell,D.Snyers. A colony of ant-like agents for partitioning in VLSI technology, in: P.Husbans,I.Harvey(Eds.), Proceeding of the Fourth European Conference on Artificial Life, MIT Press, Cambridege,MA,1997,pp.417-424