A Modified Ant-based Clustering for Medical Data

C. Immaculate Mary, M.C.A.,M.Phil.,
Associate Professor Of Computer Science,
Sri Sarada College, Salem-636016,
TamilNadu, India

Dr. S.V. Kasmir Raja
Dean (Research), S.R.M. University , Chennai,
Tamil Nadu, India.

Abstract-**Ant-based techniques, in the computer sciences, are designed for those who take biological inspirations on the behavior of the social insects. Data-clustering techniques are classification algorithms that have a wide range of applications, from Biology to Image processing and Data presentation. The ant-based clustering technique has been proven a promising technique for the data clustering problems. In this paper a modified ant-based clustering is proposed for medical data processing. The performance of the proposed method is compared with k-means clustering.**

Keywords- Clustering; k-means; ACO; validation; Entropy ; F-measure

I. INTRODUCTION

Ant-based clustering and sorting [1] was inspired by the clustering of corpses and larval sorting activities observed in real ant colonies [2]. The algorithm's basic principles are straightforward [3]: 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 picking and dropping operations are biased by the similarity and density of data items within the ants' local neighborhood: ants are likely to pick up data items that are either isolated or surrounded by dissimilar ones; they tend to drop them in the vicinity of similar ones. In this way, a clustering and sorting of the items on the grid is obtained. Hence, like Ant Colony Optimization (ACO, [4]), ant-based clustering and sorting is a distributed process that employs positive feedback. However, in contrast to ACO, no artificial pheromones are used; instead, the environment itself serves as stigmergic variable [5].

Ant-based clustering and sorting was originally introduced for tasks in robotics by Deneubourg et al. [1]. Lumer and Faieta [3] modified the algorithm to be applicable to numerical data analysis, and it has subsequently been used for data-mining [6], graph-partitioning [7,8,9] and text-mining [10,11,12].

Such ant-based methods have shown their effectiveness and efficiency in some test cases [13].

However, the ant-based clustering approach is in general immature and leaves big space for improvements. With these considerations, however, the standard ant-based clustering performs well; the algorithm consists of lot of parameters like pheromone, agent memory, number of agents, number of iterations and cluster retrieval etc. For these parameters more assumptions have been made in the previous works.

In this paper, a modified ant-based clustering is proposed. Here, the algorithm doesn't have any parameters and assumptions. And the proposed method will automatically calculate the number of ants required for clustering. With this modification a modified ant-based clustering is presented and compared with k-means clustering.

The paper is organized as follows: the following section describes the standard ant-based clustering and the proposed method, section III presents k-means clustering, section IV presents the experiments and results and the proposed work is concluded in section V.

II. ANT-BASED CLUSTERING

One of the topics that was deeply explored in the past by ethnologists was the understanding of mechanism how almost blind animals were able to find the shortest way from a nest to food. Comprehension of the way to achieve this task by nature was the first step to implement that solution in the algorithm area. Main inspirations to create ACO metaheuristic were research and experiments carried out by Goss and Deneubourg [1]. Ants (Linepithaema humile) are the insects that live in the community called colony. The primary goal of ants is the survival of the whole colony. A single specimen is not essential, only bigger community can efficiently cooperate. Ants possess the ability of such efficient cooperation. It is based on work of many creatures which evaluate one solution as a colony of cooperative agents. Individuals do not communicate directly.

Each ant creates its own solution that contributes to the whole colony's solution [14]. The ability to find the shortest way between the source of food and the ant heel is a very important and interesting behavior of the ant colony. It has been observed that ants use the specific substance called pheromone to mark the route they have already gone

through. When the first ant randomly chooses one route it leaves the specific amount of pheromone, which gradually evaporates. Next ants which are looking for the way, will, with greater probability, choose the route where they feel more pheromone and after that they leave their own pheromone there. This process is autocatalic – the more ants choose a specific way, the more attractive it stays for the others. The above information comes mainly from the publications by Marco Dorigo. He is the one who most of all contributed to development of the research in the ant systems area. His publications are the largest repository of ACO information [14, 15].

a. The Standard Method

In this section we analyze the general ideas of the ant-based data clustering technique originated by Deneubourg et. al. [1]. In their work, a model was developed to mimic the "clustering" behavior for the Messor sancta ants to clean the nests by piling different sorts of items (corpuses, larva, and foods) in different positions. A simple mechanism guides the ants to complete this task: when an ant encounters an item, it tends to pick the item up if the item is dissimilar with the surrounding items; later, if the same ant moves to another position that contains a variety of items that is of the same type of the item being carried by the ant (e.g. the ant carries a dead body to a place that holds a good number of dead bodies), the ant would probably drop the item to that position. With such mechanism, as all the ants in a nest repeat such activities for some period of time, it can be expected that some clusters may be formed with each cluster being comprised of the same type of items.

In Deneubourg et. al.'s model, the prior ant-colony behavior is imitated to perform data clustering. In general, an ant-based clustering algorithm based on Deneubourg et. al.'s model can be described as follows. It first assumes the data objects or items to be clustered are randomly laid down on a two-dimensional m × m grid or clustering workspace, where m depends on the number of items. Each cell in the grid can contain at most one item. A few artificial "ants" are also placed in the same grid at random. At this initial stage, each ant does not "carry" any item. After completing such initialization process, a cyclic process is designed in which each ant sequentially conducts the following three activities at each step:

- Picking up: At current step, if the ant does not carry any item (i.e. the ant is an "unladen" ant), and if it "encounters" an item o_i (i.e. the ant and the item are located in the same cell at the current step), the ant decides to pick up or ignore that item according to a "picking up" probability P_p, which is a function of local density that determines the similarity of the item o_i with its neighboring items. Less similar items are present, more probably the ant picks the item up.

- Moving: After making the "picking up" decision, the ant randomly moves from the current cell to another cell in the grid. In some variations of the Deneubourg-style ant clustering methods, the ant can only move to an adjacent cell that is not occupied by another ant; but in some other variations, the ant can move across any distance to any other unoccupied cell in the grid.

- Dropping: When the ant reaches a new cell, and if it carries some item (i.e. it is a "laden" ant), the ant requires making another decision whether or not dropping the laden item to this cell, in case that this arrived position does not occupied by another item. Again, the ant calculates another probability (called dropping probability, P_d), which is another function of the similarity between the laden item with the items neighboring this newly-arrived cell. More similar items exist in a local area around the cell, more likely the ant drops the item.

Repeating such activities, the ant may gradually split different types of items into different clusters. The overall process ends when the clusters become stable or the maximal running iteration is reached. Obviously, the key factors of the above ant clustering algorithm are the picking up and dropping probability functions Pp (1)and Pd.In Deneubourg et. al.'s model, these two functions are determined by defined as the following equations:

$$\text{Picking up probability, } P_p = \left(\frac{k_1}{k_1 + f} \right)^2 \quad (1)$$

$$\text{Dropping probability, } P_d = \left(\frac{f}{k_2 + f} \right)^2 \quad (2)$$

$f = f(o_i)$ is a similarity or relevance measure of the item o_i in its neighborhood, while k_1 and k_2 are threshold constants (picking-up threshold and dropping threshold, respectively). P_p is high when f = 0, indicating that the item o_i would be picked up with a high probability if o_i is dissimilar with its surrounding items. Oppositely, P_d is high when the value if f is high, indicating o_i would probably be dropped to a cell where there are quite some items similar with this item o_i nearby.

Equation (3) is used as the similarity measure between a specific data item (denoted by o_i, following the prior description) and the neighboring data in the grid of the aforementioned ant-based clustering algorithm is:

$$f(o_i) = \frac{\sum_{o_j} sim(o_i, o_j)}{N(o_i)} \quad (3)$$

Where N is the number of cells neighboring the cell where the data item o_i is going to be picked up from or dropped to (of course, the definition of neighborhood may be somewhat different in different variations of the ant-based clustering technique). The above equation indicates an average similarity between the data item o_i and all its neighboring data items.

b. The Modified Ant-based Clustering

In this modified ant-based clustering algorithm, the similarity measures that is the cosine distance between each data items are calculated initially and they are normalized. This domain is considered as cluster space for ant-based clustering. With this domain, an single agent that is the first ant is placed at random data item, then it search for its neighbor at random again. Once it finds any neighbor which is not occupied are picked by any other ant, then for that neighbor the probability of pick and drop is calculated. Based on these probabilities the decision is made whether to choose the data item or to drop it. If the data item is selected then the index is marked with current ant number. If it is not then it is stored with dropped index (0.5). Then it moves to the next neighbor. This routine is repeated till it could not find any other similar data item. The picking up threshold k1 is set to 0.2; and the dropping threshold k2 is set to 0.05. Once a run is over for an agent, then the cluster space is checked for uncovered data items. If we could find any uncover data item then the next ant is introduced ant finds its cluster as similar procedure. This entire procedure is repeated till there is no uncovered data item. The overall procedure of the proposed algorithm can be described as follows:

Algorithm

procedure for modified-ant-based-data-clustering

Calculate the similarities of the data item.
Place the data items in the cluster-space at random position.
Initialize the cluster index for the entire data item with 0.
Initialize the cluster index with 1.
Introduce an ant
Do
Initialize the ant by choosing a data item randomly and place the ant.
Assign the current cluster index
for each data item do
if the data item is uncovered & not dropped
Based on the similarity measure calculate the pick & drop probability
if pick > drop then
Add the data item with the current cluster
And assign the current cluster index
Move to the next neighbor.
else
Assign the drop index (0.5)
// used to restrict the ant from choosing the same data-item again.
end
end
end-for
for each dropped data item do
Change the cluster index back to 0.
end-for
if any uncovered data items in the cluster-space
Increase the cluster index by 1.
Introduce the next ant
repeat
else
break
end if
repeat

end-procedure

III. K-MEANS CLUSTERING

The first algorithm we compare against is the well-known k-means algorithm. Starting from a random partitioning, the algorithm repeatedly (i) computes the current cluster centers (i.e. the average vector of each cluster in data space) and (ii) reassigns each data item to the cluster whose centre is closest to it. It terminates when no more reassignments take place. By this means, the intra-cluster variance, that is, the sum of squares of the differences between data items and their associated cluster centers is locally minimized. k -means' strength is its runtime, which is linear in the number of data elements, and its ease of implementation. However, the algorithm tends to get stuck in suboptimal solutions (dependent on the initial partitioning and the data ordering) and it works well only for spherically shaped clusters. It requires the number of clusters to be provided or to be determined (semi-) automatically. In our experiments, we run k-means using the correct cluster number .

IV. RESULTS AND EXPERIMENTS

For clustering, two measures of cluster "goodness" or quality are used. One type of measure allows us to compare different sets of clusters without reference to external knowledge and is called an internal quality measure. As mentioned in the previous section, we will use a measure of "overall similarity" based on the pairwise similarity of data

items in a cluster. The other type of measures lets us evaluate how well the clustering is working by comparing the groups produced by the clustering techniques to known classes. This type of measure is called an external quality measure. One external measure is entropy [16], which provides a measure of "goodness" for un-nested clusters or for the clusters at one level of a hierarchical clustering. Another external measure is the F-measure, which, as we use it here, is more oriented toward measuring the effectiveness of a hierarchical clustering. The F measure has a long history, but was recently extended to data item hierarchies in [17].

There are many different quality measures and the performance and relative ranking of different clustering algorithms can vary substantially depending on which measure is used. However, if one clustering algorithm performs better than other clustering algorithms on many of these measures, then we can have some confidence that it is truly the best clustering algorithm for the situation being evaluated. As we shall see in the results sections, the bisecting k-means algorithm has the best performance for the three quality measures that we are about to describe.

a. Entropy

We use entropy as a measure of quality of the clusters (with the caveat that the best entropy is obtained when each cluster contains exactly one data point). Let CS be a clustering solution. For each cluster, the class distribution of the data is calculated first, i.e., for cluster j we compute p_{ij}, the "probability" that a member of cluster j belongs to class i. Then using this class distribution, the entropy of each cluster j is calculated using the standard formula

$$E_j = -\sum_i p_{ij} \log(p_{ij}) \quad (4)$$

where the sum is taken over all classes. The total entropy for a set of clusters is calculated as the sum of the entropies of each cluster weighted by the size of each cluster:

$$E_{CS} = \sum_{j=1}^{m} \frac{n_j * E_j}{n} \quad (5)$$

where n_j is the size of cluster j, m is the number of clusters, and n is the total number of data points.

b. F measure

The second external quality measure is the F measure [17], a measure that combines the precision and recall ideas from From the table 1 we can infer that the validity measure entropy tends to be decreased and F measure increased. Our aim is also to minimize the entropy and maximize the F measure. So, this validation shows that Modified ACO is more effective than standard ACO.

information retrieval [18]. We treat each cluster as if it were the result of a query and each class as if it were the desired set of data items for a query. We then calculate the recall (6)and precision (7) of that cluster for each given class. More specifically, for cluster j and class i

$$\text{Recall}(i, j) = nij / ni \quad (6)$$

$$\text{Precision}(i, j) = nij / nj \quad (7)$$

where n_{ij} is the number of members of class i in cluster j, n_j is the number of members of cluster j and ni is the number of members of class i.

The F measure of cluster j and class i is then given by

$$F(i, j) = (2 * \text{Recall}(i, j) * \text{Precision}(i, j)) / ((\text{Precision}(i, j) + \text{Recall}(i, j)) \quad (8)$$

For an entire hierarchical clustering the F measure of any class is the maximum value it attains at any node in the tree and an overall value for the F measure is computed by taking the weighted average of all values for the F measure as given by the following. Equation (9)

$$F = \sum_i \frac{n_i}{n} \max \{F(i, j)\} \quad (9)$$

where the max is taken over all clusters at all levels, and n is the number of data items.

	Wisconsin Breast Cancer Dataset			**Dermatology Dataset**		
	K-Means	***ACO***	***Modified ACO***	***K-Means***	***ACO***	***Modified ACO***
No. of Classes	2	2	2	6	6	6
No. of Clusters	2	2	2	6	6	6
Entropy	0.2373	0.0633	0.0029	0.0868	0.0524	0.0051
F measure	0.9599	0.9872	0.9986	0.8303	0.9645	0.9958

The following table presents the results of the calculated measures for each algorithm.

TABLE 1. Performance of clustering algorithms on biomedical dataset

V. CONCLUSION

The experiments confirm an argument that the ant algorithms can be successfully implemented in data items processing. The attempt of creating valuable clustering method based on modified ACO meta-heuristic was success.

This proves the universal nature and flexibility of ACO meta-heuristic. The results obtained during experiments are characterized by good quality, speed for big collections of data items and flexibility in determining the number of resultants groups.

REFERENCES

.

[1] Deneubourg J.L., Goss S., Franks, N. Sendova-Franks A., Detrain C., and Chétien L. The Dynamics of Collective Sorting: Robot-like Ants and Ant-like Robots, In Proceedings of the 1st International Conference on Simulation of Adaptive Behaviour: From Animals to Animats,. MIT Press, Cambridge, MA, USA, 1:356-363, 1991.

[2] Bonabeau E., Dorigo M., and Theraulaz G. Swarm Intelligence – From Natural to Artificial Systems. Oxford University Press, New York, 1999.

[3] Lumer E., and Faieta B. Diversity and adaptation in populations of clustering ants. In Proceedings of the Third International Conference on Simulation of Adaptive Behaviour: From Animals to Animats. MIT Press, Cambridge, MA, 3:501–508, 1994.

[4] Dorigo M., and Di Caro G. Ant Colony Optimization: A new meta-heuristic. In D. Corne, M. Dorigo, and F. Glover, editors, New Ideas in Optimization, pages 11–32. McGraw-Hill, London, UK, 1999.

[5] Dorigo M., Bonabeau E., and Theraulaz G. Ant algorithms and stigmergy. Future Generation Computer Systems, 16(8):851–871, 2000.

[6] Lumer E., and Faieta B. Exploratory database analysis via self-organization, 1995.

[7] Kuntz P., and Snyers D. Emergent colonization and graph partitioning. In Proceedings of the Third International Conference on Simulation of Adaptive Behaviour: From Animals to Animats. MIT Press, Cambridge, MA, 3:494–500, 1994.

[8] Kuntz P., and Snyers D. New results on an ant-based heuristic for highlighting the organization of large graphs. In Proceedings of the 1999 Congress on Evolutionary Computation, IEEE Press, Piscataway, NJ, 1451–1458, 1999

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

[10] Handl J., and Meyer B. Improved ant-based clustering and sorting in a document retrieval interface. In Proceedings of the Seventh International Conference on Parallel Problem Solving from Nature, Springer-Verlag, Berlin, Germany, 2439:913–923, 2002

[11] Hoe K., Lai W., and Tai T. Homogeneous ants for web document similarity modeling and categorization. In Proceedings of the Third International Workshop on Ant Algorithms, Springer-Verlag, Heidelberg, Germany, 2463:256–261, 2002.

[12] Ramos V., and Merelo JJ. Self-organized stigmergic document maps: Environments as a mechanism for context learning. In Proceedings of the First Spanish Conference on Evolutionary and Bio-Inspired Algorithms, Centro Univ. M´erida, M´erida, Spain, 284–293, 2002.

[13] Handl J., Knowles J., and Dorigo M. "On the Performance of Ant-based Clustering", In Design and Application of Hybrid Intelligent Systems, Frontiers in Artificial Intelligence and Applications,. Amsterdam, the Netherlands: IOS Press, 104:204-213, 2003.

[14] Dorigo M., Maniezzo V., Colorni A., The ant systems: optimization by colony of cooperating agents, IEEE Transactions on Systems, Man, and Cybernetics-PartB, 1996.

[15] Dorigo M., Optimization, Learning and Natura Algorithms (In Italia), PhD thesis Dipartimento di Elettronica e Informazione, Politecnico di Milano, IT, 1992.

[16] Shannon CE., A mathematical theory of communication, Bell System Technical Journal, 27:379-423 and 623-656, July and October, 1948

[17] Larsen B., and Aone C. Fast and Effective Text Mining Using Linear-time Document Clustering, KDD-99, San Diego, California, 1999.

[18] Kowalski G, Information Retrieval Systems – Theory and Implementation, Kluwer Academic Publishers, 1997.