Annales UMCS Informatica AI 3 (2005) 315-323

Annales UMCS
Informatica
Lublin-Polonia
Sectio AI

http://www.annales.umcs.lublin.pl/

ACO-based document clustering method

Łukasz Machnik[*]

Department of Computer Science, Warsaw University of Technology,
Nowowiejska 15/19, 00-665 Warsaw, Poland

Abstract

Ant systems are flexible to implement and give possibility to scale because they are based on multi agent cooperation. The aim of this publication is to show the universal character of that solution and potentiality in implementing it in wide areas of applications. The increase of demand for effective methods of large document collections management is a sufficient stimulus to place the research on the new application of ant based systems in the area of text document processing. Hitherto existing far generated ant based clustering methods are presented and briefly described at the beginning of that article. Next, the author defines the ACO (Ant Colony Optimization) meta-heuristic, which was the basis of the method developed by him. Presentation of the details of the ant based documents clustering method is the main part of publication.

1. State of research on ant-based clustering methods

Ant based algorithms are assigned to the group of multiagent systems. In such systems single agent (artificial ant) behavior is inspired by behavior of real ants.

Ant based clustering and sorting algorithm

Ant based clustering and sorting algorithm was first introduced by Deneubourg in 1990 [1]. As its name implies, two types of natural ant behavior are modeled by this algorithm. Firstly, clustering, where ants gather items to form heaps. An example for this is the clustering of dead corpses (cemetery formation) observed in the species of *Pheidole pallidula.* Secondly, sorting, where ants discriminate different kinds of items and spatially arrange them according to their properties. This type of activity can be observed in nests of *Leptothorax unifasciatus*, where larvae are arranged dependent on their sizes. In the Deneubourg's model ants are modeled by simple agents, which randomly move in their environment, which is a square grid with periodic boundary conditions. Data items that are scattered within this environment can be picked

[*] *E-mail address*: L.Machnik@ii.pw.edu.pl

up, transported and dropped by the agents. The picking and dropping operations of each individual agent are based on the probabilities.

$$p_{pick}(i)=\left(\frac{k^{+}}{k^{+}+f(i)}\right)^{2}, \tag{1}$$

$$p_{drop}(i)=\left(\frac{f(i)}{k^{-}+f(i)}\right)^{2}. \tag{2}$$

In the above formulas, *f(i)* is an estimation of the fraction of data items in an ant's immediate environment, that are similar to the data item the ant currently considers to pick up or drop respectively. The parameters k^{+}, k^{-} determine the influence of the neighborhood function, and basically were fixed to 0.3 and 0.1 respectively. A few form of neighborhood function exist. At present the most popular is the formula presented by Lumer and Faieta [2].

$$f(i)=\begin{cases}\frac{1}{\sigma^{2}}\sum_{j}\left(1-\frac{\delta(i,j)}{\alpha}\right) & if\ f(i)>0\\ 0 & otherwise\end{cases} \tag{3}$$

In the above formula, $\delta(i,j)\in[0,1]$ stands for the dissimilarity function defined between points in the data space, $\alpha\in[0,1]$ is a data-dependent scaling parameter, and σ^{2} is the size of the local neighborhood (typically $\sigma^{2}\in\{9,25\}$). The agent is located in the centre of this neighborhood; the radius of perception in each direction is therefore $(\sigma-1)/2$. During the calculation of neighborhood function, in the process of counting, there is a j number of elements in actual perception of the agent considered.

AntTree

The above introduced ant-based clustering and sorting algorithm is a well known solution and, till recently, one of the ant-based solutions area associated with the clustering problem. Recently, the attempts of working out the new solutions in ant systems area have been observed [3,4]. The solution introduced by Azzag and Venturinieg is an attempt to adopt a new biological model in computer science. The origin of proposed solution is the observation of ants of *Linepithema humiles* and *Oecophylla langinoda* species. Those individuals have the ability to discover tree structures of dataset, through building live constructions using their own bodies [5]. Two types of chains can be created by ants. The first created while crossing empty space and the second created during the nest building process. Constructions made by ants can be interpreted in many different ways, also as hierarchical clustering of data or visualization of dataset.

Below, the basic rules of that algorithm, called by the authors AntTree, will be presented. Each ant represents a single node of the constructed tree, which is an element that takes part in the clustering process. Attaching next elements to the tree begins always from a fictitious node that can be treated as a root of the tree. Next, ants can directly connect to the root or move through the structure created by bodies of others individuals and finally join to them. The decision to move or connect is made by an ant, based on similarity function calculation in its local neighborhood. The results presented by the authors confirm effectiveness of that method. However, the details of introduced method have not been published yet.

2. The origin of ACO (Ant Colony Optimization)

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 inspiration to create ACO metaheuristic were research and experiments carried out by Goss and Deneubourg [6]. 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 [8]. 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 [7,8].

3. ACO-based clustering method

The analogy between finding the shortest way by ants and finding documents is obvious (the shortest way between documents). In addition ability to use agents constructing their individual solutions as an element of the general solution, became the stimulus to begin research on using the ant based algorithms in the documents clustering proccess.

Adopting the ACO concepts to documents clustering task

For the needs of building an effective method of classifying the documents, it is necessary to make a choice of possible modification and adjusting of the concepts specific to real ants, so could be effectively used to solve the problems connected with text mining,

– A colony of co-operating individual specimen.

Artificial ants build a solution by moving along the graph of a problem, from one document to the other. During each iteration *m* number of ants constructs a solution in *n* number of steps, using a probabilistic law of making a decision. In practice, when visiting a specific document *i* ant chooses the next document *j* to move to, a pair (i, j) is added to the solution constructed at the moment. This step is repeated until the ant builds a complete solution for the specific iteration. Considering the fact that this version of the algorithm is serial, after each ant finds a solution in a specific iteration process of leaving of certain amount of pheromone associated with a pair of documents follows. After that the ant dies. Yet new ants appear in her place, whose goal is to find a solution in the following iteration, leave a pheromone and die. The pattern repeats until gaining the best result, or until performing a specific amount of iterations.

– A pheromone trace and its force to influence

From available variants of leaving pheromone on the path, the author chose a partial variant. The ants leave a pheromone in a specific amount which equals a quotient of a constant and a length of a found path. In addition the decay of the pheromone follows after constructing of all partial solutions – the sum of distances between all of visited documents. The communication pheromone path is being changed while finding a solution to a problem just to show the experience gained by ants while solving the problem.

– Finding the shortest path

Co-ordinate describing the location of the specific document in space will be a vector representing the frequency of words occurring in the document. To describe the distance between the documents a simple measure in multidimensional space will be used – cosine distance. Finding the shortest path will be represented by finding such a sequence of passing from one document to the other, that the sum of the reverse of cosine distances between the following elements of the examined set would be smaller. The use of the reverse of cosine distance is necessary because the increase of cosine distance evidences on greater similarity between documents.

– Accidental movement of individual ants in the starting phase of finding the path

Maintaining this condition is necessary because in the starting phase of algorithm action the ants are not able to use the experience of their predecessors. The pheromone trace between individual documents is equal to the selected constant value. Such a situation forces fully accidental choice of the documents in the starting phase of finding the path.

– Artificial ants live in the artificial, discreet world and can move only from one to the other specific position – states of the discreet world

The set of states between which agents can move will be defined as a set of vectors representing the individual documents. As we assumed earlier, each document will be represented by a vector based on frequency of appearance of the specific worlds in examined text.

– The amount of the pheromone left by the artificial ant is connected with quality function of so far achieved solution

The amount of the pheromone left by ants is proportional to the quality of the solution they find: the shorter is the distance between the documents, the bigger is the amount of the pheromone left on the pairs of the documents – documents used to create the solution. The issue that still cannot be forgotten is requirement to evaporate the pheromone. It is also necessary to exclude the stagnation phenomenon, which means choosing the same route by all ants too early.

– Past states memory

The artificial ants are equipped with the memory of passed states, which is supposed to prevent the multiple location of one ant in the same position (it is necessary because there is a possibility/danger that ants could fall into cycles, which could make finishing the building of the solution impossible)

Details of processing

The method of document clustering which is introduced here, is based on artificial ant system. Application of such a solution will be used as a method of finding the shortest path between the documents, which is the goal of the first phase (trial phase) of the considered method. The second phase (dividing phase) will have a task to actually separate a group of documents alike. The aim of trial phase is to find the shortest path connecting every document in the set using ACO algorithm. That is equivalent to building a graph, whose nods would make up a set of analyzed documents. The probability of choosing next document *j* by ant *k* occupying document *i* is calculated by the following function.

$$p_{ij}^{k}(t)=\frac{\left[\tau_{ij}(t)\right]^{\alpha}*\left[s_{ij}\right]^{\beta}}{\sum\limits_{k\in Z_k}\left[\tau_{ik}(t)\right]^{\alpha}*\left[s_{ik}\right]^{\beta}}. \tag{4}$$

In the above formula, Z_k represents list of documents not visited by ant *k*, $\tau_{ij}(t)$ represents the amount of pheromone trail between documents *i,j*, α is the intensity of pheromone trail parameter, β is the visibility of documents parameter, however s_{ij} is the cosine distance between documents *i* and *j*. After ants complete their trace the pheromone trail is evaporated and new amount of pheromone is left between every pair of documents. The amount of pheromone that is left by the ants is dependent on the quality of the constructed solution (length of the path). In practice, adding the new portion of pheromone to trail and its evaporating is implemented by the formula presented below. This formula is adapted to every pair of documents (*i,j*).

$$\tau_{ij}(t)\ \leftarrow\ (1-\rho)\cdot\tau_{ij}(t)+\Delta\tau_{ij}(t). \tag{5}$$

In the above formula, $\rho\in(0,1>$ stands for the pheromone trail decay coefficient, while $\Delta\tau_{ij}(t)$ is an increment of pheromone between documents (i,j). Below the dependence that controls the amount of pheromone left by ant k between the pair of documents (i,j) is presented.

$$\Delta\tau_{ij}^{k}(t)=\begin{cases}\dfrac{n}{L_k(t)} & dla\ (i,j)\in T^k(t)\\ 0 & dla\ (i,j)\in T^k(t)\end{cases} \tag{6}$$

In the above formula, $T^k(t)$ means a set of document pairs that belong to the path constructed by ant k, $L_k(t)$ is the length of path constructed by ant k, while n is the number of all documents. Finding the shortest path connecting every document in the set will be equivalent to building a graph, whose nods would make up a set of analyzed documents. Documents alike would be neighboring nods in the graph, considering that the rank of the individual nods will fulfill the condition of being smaller or equal to 2, which means that in the final solution one of the documents would be connected to only two others (similar) – each document in the designed solution would appear only ones. Gaining such a solution would mean the end of the first phase, known as *preparing*. In the following stage of the process it is necessary to separate a group of documents alike in a sequence obtained in the first phase. The separation of groups is obtained by appropriate processing of the sequence of documents (the shortest path) received in the preparing phase. The following individual steps of that process are described. The vector that represents the the first document in sequence is recognized as centroid μ of the first group that is separated. Using the whole similarity measure $\|\mu\|^2$, in which the length of the centroid vector that represent considered group, square, is a measure of considered group cohesion, we calculate the cohesion of the first group. In the next step the next document from the sequence is added, to the first group. The centroid μ of that group and a change of group cohesion after adding new element are calculated. If the change of group cohesion has an acceptable value (smaller than the predefined value of γ parameter), then the considered element permanently becomes the member of first group and we try to enlarge this group by adding the next element from the sequence.

$$\left|\Delta\|\mu\|^2\right| < \gamma . \tag{7}$$

However, if the change of a group cohesion after adding the considered element does not have an acceptable value (higher than the predefined value of γ parameter), then the separation of the first group is finished and the separation of the next (second) group begins. The vector of the considered document that could not be added to the first group becomes an initial centroid of the new group. The whole process is repeated from the beginning. Processing is finished when the whole sequence of documents is done.

Method variants

The amount of separated groups depends precisely on admissible change of group cohesion. By small admissible change of group cohesion as a result of processing we received a large number of groups with a high degree of cohesion. The increase of admissible change of group cohesion causes receiving a smaller number of groups with less cohesion. In connection with the above conclusion it is possibile to propose two variants of the considered method. The first variant called by author – single pass, is based on very precise execution of the trial phase – a lot of ants. The duration of the first phase increases, however, this activity permits to accept higher change of group cohesion during dividing phase and finishing processing after single pass of algorithm – single trial phase and single dividing phase. The clustering method that uses the single pass variant is the example of non-hierarchical clustering method. The main advantage of that method is that operator does not have to set the expected number of clusters at the beginning of processing. The results received in this variant are less precise than those from the second variant, however, the time of processing is much shorter than the time of the second proposed variant. This type of considered method can also act as a trial phase for other clustering algorithms. The example can be separations of centroids for K-means method.

The second variant called by the author – periodic, differs a little bit from the variant proposed earlier. It assumes periodic processing of both phases: trial and dividing. In every iteration of dividing phase the small numbers of neighbours are connected into small groups. The acceptable value of cohesion change is very small in initial phases and is gradually increased to allow group creation in next iterations. Each group during processing is represented by centroid. After group creation and centroids calculations the next iteration can be started – finding the shortest path between centroids and documents. The whole process is finished when all documents are connected as a single cluster or when the stop criterion is reached. This variant is an example of agglomerative hierarchical clustering method that begins from a set of individual elements which are then connected to the most similar elements forming bigger and bigger clusters. The result of hierarchical technique processing is creating a nested sequence of partitions. The main partition is placed at the top of hierarchy. It includes all elements from the considered collections. The base of hierarchy creates individual elements. Every middle level can be represented as combination of clusters that are at the lower level in hierarchy. The user can choose any level that satisfies him as solution.

4. Summary

Clustering of big document sets may be classified as a complicated computing problem. Ant systems are flexible to implement and give possibility to scale because they are based on multi agent cooperation. Experiments have shown that the idea of using ACO meta-heuristic in the described way to solve the document clustering problem is useful and functional. Indication of the place of that method in the document clustering area needs comparison with other available methods. The research is in progress and detailed results of tests will be published soon.

References

[1] Deneubourg J.-L., Goss S., Franks N., Sendova-Franks A., Detrain C., Chretien L., *The dynamics of collective sorting: Robot-like ants and ant-like robots*, First International Conference on Simulation of Adaptive Behaviour: From Animals to Animats 1, MIT Press, MA, (1991) 356.

[2] Lumer E., Faieta B., *Diversity and adaptation in populations of clustering ants*, Third International Conference on Simulation of Adaptive Behaviour: From Animals to Animats 3, MIT Press, (1994) 501.

[3] Machnik Ł., *Documents Clustering Techniques*, IBIZA 2004, Poland, (2004).

[4] Azzag H., Venturini G., *A clustering model using artificial ants*, Universite Francois-Rabelais, France, (2004).

[5] Lioni A., Sauwens C., Theraulaz G., Deneubourg J.-L., *The dynamics of chain formation in Oecophylla longinoda*, Journal of Insect Behavior, 14 (2001).

[6] Deneubourg J. L., Pasteels J. M., Verhaeghe J. C., *Probabilistic behaviour in Ants: a strategy of errors*, Journal of Theoretical Biology, (1983) 259.

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

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