

## A Text Categorization Using Swarm Intelligence



### Engineering

**KEYWORDS :** Swarms, clustering, Ant Algorithm, Ant-clustering, Text-clustering

**DURGA BHAVANI  
DASARI**

Research Scholar, Dept of CSE, Jawaharlal Nehru Technological University, Hyderabad, India

**Dr. VENU GOPALA  
RAO.K**

Professor, Dept of CSE, G. Narayanamma Institute of Technology and Science, Hyderabad, India

### ABSTRACT

*This paper presents an algorithm for the automatic grouping of PDF documents, and with potential application for Web document classification. The algorithm developed is based on an ant-clustering algorithm, which was inspired by the behaviour of some antspecies in the organization their nests. To apply the ant clustering algorithm for text document classification, two modifications had to be introduced in the standard algorithm: 1) the use of a metric to evaluate the similarity degree of text data, instead of numeric data; and 2) the proposal of a cooling schedule for a user-defined parameter so as to improve the convergence properties of the algorithm.*

### INTRODUCTION

Until very recently, obtaining information about a given subject involved going to a library or university and searching for the desired contents. All bibliographical resources of a library (e.g. books, journals, magazines and newspapers) are grouped by indices, i.e., collections of terms that point to the sites where they can be found. These terms can be the names of the authors, the subjects, the year of publication, and so forth. With time, not only the number of libraries increases, but also the amount of information available. To minimize this problem, information retrieval systems have been developed and widely used in libraries, universities, companies and all other places where information resources have to be stored and consulted. Information retrieval systems are aimed at helping the storage of new information resources and speeding up the search for a specific subject [1],[2].

The Internet has emerged as one of the most important information resources, in most cases of public use, available nowadays. This can be easily observed by the broad number of digital libraries in the Web [3]. As is the case with the "physical" libraries, digital or virtual libraries also suffer from the difficulties in organizing and searching for information. Although information retrieval systems have contributed significantly to the organization and retrieval of information, the success of the system still depends on maintenance, because one has to be responsible for taking the new information resources, indexing and cataloguing them. These processes are tedious and time consuming.

This paper presents a system for the automatic organization of digital documents in PDF format. The approach is based on an ant-clustering algorithm proposed by Lumer and Faieta [4] as a development of the ideas introduced by Deneubourg et al.[5]. This method was designed as part of an academic virtual community currently under development. This community is characterized as a scientific paper collection (PDF files) automatically classified and stored in folder structures of a server and in which academics are able to exchange experience and knowledge.

This paper is organized as follows. Section 2 briefly introduces swarm intelligence and the ant-clustering algorithm. Section 3 describes how the algorithm was implemented to solve text clustering, and Section 4 presents some simulation results. Conclusions and some proposals for future works are provided in Section 5.

### SWARMS, ANTS AND CLUSTERING

The social behaviour of ants has attracted the interest of researchers in many different disciplines, from the biosciences to computer science and engineering. One emerging field of investigation that has been increasingly receiving attention over the past years is the so-called biologically inspired computing [6], in particular, swarm intelligence [7],[8]. The term swarm intelligence was coined in the late 1980's to refer to cellular robotic systems in which a collection of simple agents in an environment interact according to local rules [9],[10].

Two main lines of research can be identified in swarm intelligence: 1) the works based on social insects [7]; and 2) the works based on the ability of human societies to process knowledge [8]. Although the resultant approaches are quite different in sequence of steps and sources of inspiration, they present some commonalities. In general terms, both rely on a population (colony or swarm) of individuals (social insects or particles) capable of interacting (directly or indirectly) with the environment and each other. As a result of these interactions there may be a change in the environment and/or in the individuals, what will lead to useful emergent phenomena.

Among the many social behaviours of ants, researchers have registered the way some ant species work collaboratively in the task of grouping dead bodies so as to keep the nest clean [11],[12]. After placing corpses of ants randomly in a certain environment, it can be observed that, with time, the ants tend to cluster all dead bodies in specific regions of the environment thus forming piles of dead bodies. The first ant clustering algorithm inspired by this clustering behaviour of ants was introduced in [5], where a population of robots had to group together objects without any central control.

Lumer and Faieta [12] adapted the robots ant-clustering algorithm for the analysis and classification of numerical data, thus introducing the standard ant-clustering algorithm (ACA). Since its proposal, in 1994, the ACA has passed through some modifications and has been applied to several domains, from data mining [12], to graphpartitioning [13]-[15], to text-mining [16]-[18]. Independently of the application domain and particular version of the algorithm, ant-clustering algorithms based on ACA follow a set of basic, general principles.

Given an input data set composed of  $N$   $l$ -dimensional vectors to be clustered, these data are spread all over a bi-dimensional (toroidal) grid of size  $m \times m$ . Actually, the data themselves are not spread over the grid, only some sort of indices that indicate

where a given object is placed. A colony of ants (agents) is allowed to move on the grid picking up, carrying and dropping off objects based on some probabilistic rules. The movement of an ant is characterized by its displacement in a grid cell in any direction adjacent to its current position. The ants can perceive a neighbourhood in the environment, the most common one being a square neighbourhood of size 3x3. In the beginning of the iterative process of adaptation, objects and ants are randomly placed on the grid.

The ants then start moving randomly on the grid. If an ant is not carrying an object and finds an object *i* in its neighbourhood, it picks up this object with a probability that is inversely proportional to the number of similar objects in the neighbourhood, as described in Eq. (1).

$$P_{pick}(i) = \left( \frac{k_p}{(k_p + f(i))^2} \right) \tag{1}$$

If, however, the ant is carrying an object *i* and perceives a neighbour cell in which there are other objects, then the ant drops off the object it is carrying, with a probability that is directly proportional to the object's similarity with the perceived ones, as described in Eq. (2).

$$P_{drop}(i) = \left( \frac{f(i)}{(k_d + f(i))^2} \right) \tag{2}$$

The parameters *k<sub>p</sub>* and *k<sub>d</sub>* are, respectively, the picking and dropping constants that weighs the influence of the function *f(i)* on the picking and dropping probabilities. Function *f(i)* provides an estimate of the density and similarity of elements in the neighbourhood of object *i*. The pseudo code presented in Algorithm 1 summarizes the standard ant-clustering algorithm (ACA).

```

procedure ACA (max_it, kp, kd)
place every item i on a random cell of the grid
place every ant k on a random cell of the grid
unoccupied by ants
t ← 1
while t < max_it
for i = 1 to N do, // for every ant
if unladen ant AND cell occupied by item xi,
then
compute f(xi) and pp(xi)
pick up item xi with probability pp(xi)
else if ant carrying item xi AND cell empty, then
compute f(xi) and pd(xi)
drop item xi with probability pd(xi)
end if
    
```

```

move to a randomly selected neighbouring
and unoccupied cell
end for
t ← t + 1
end while
print location of items
end procedure
    
```

Algorithm 1: Standard ant-clustering algorithm.

The authors [4] suggested the following function *f(i)* as the local density of items in the neighbourhood of object *i*:

$$f(i) = \begin{cases} \frac{1}{s^2} \sum_{j \in \text{Neigh}_{(s,s)}(i)} \left[ 1 - \frac{d(i,j)}{\alpha} \right] & \text{if } f > 0 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where *f(i)* is a measure of the average similarity of object *I* with another object *j* in the neighbourhood of *i*, *α* is a factor that defines the scale of dissimilarity, and *d(i,j)* is the distance between two items in their original *l*-dimensional space. Parameter *α* determines when two items should or should not be located next to each other. For instance, if *α* is too large, there is not much discrimination between two items, leading to the formation of clusters composed of items that should not belong to the same cluster; and vice-versa.

**ANT ALGORITHM FOR TEXT CLUSTERING**

To cluster PDF files, it is first necessary to convert them into text documents. Then, they have to be transformed into collections of words that will represent an object on the grid. This transformation is automatically obtained through the calculation of the relative frequency of a word in the documents (Eq. (4)). Let *f<sub>j</sub>(w)* corresponds to the number of times word *w* appears in document *j*, i.e., the frequency of word *w* in document *j*. Then, *F<sub>j</sub>(w)* represents the relative frequency of word *w* in all documents:

$$F_j(w) = \frac{f_j(w)}{\sum_v f_j(v)}; \quad v \neq w \tag{4}$$

**Note that  $0 < F_j(w) < 1$  and  $\sum_w F_j(w) = 1$ .**

This normalization serves the purpose of disregarding the number of words in the document and, instead, measure the relative importance of a word compared to the other words contained in the same document.

Instead of using the Euclidean distance as a measure of dissimilarity, the cosine measure [19], which determines the similarity between two vectors independently of their magnitude, is going to be used. One vector represents the set of words extracted from the document being carried by the ant, and the other vector represents the collection of words extracted from the documents in the neighbourhood of the ant. Eq. (5) returns the cosine of the angle between these two vectors. The cosine is equal to 1 when the vectors point in the same direction, and zero when they form a 90 degrees angle.

$$sim(D_p, D_q) = \frac{\sum_{k=1}^N F_{Dk} F_{Qk}}{\sqrt{\sum_{k=1}^N F_{Dk}^2 \sum_{k=1}^N F_{Qk}^2}} \tag{5}$$

where F<sub>Dk</sub> is the frequency of word k in the set of words extracted from the document being carried by ant D, and F<sub>Qk</sub> is the relative frequency of word k in document Q that exists in the neighbourhood of the ant.

Thus, each document is transformed into an object whose structure is an l-dimensional vector, x<sub>1</sub>x<sub>2</sub>...x<sub>l</sub> which corresponds the relative frequencies of the relevant words extracted. Table I illustrates the representation of each object.

**Table I – Objects that represent documents.**

Object	x <sub>1</sub>	...	x <sub>l</sub>
O <sub>1</sub>	0.123	...	0.232
...	...	...	...
O <sub>N</sub>	0.012	...	0.156

After the generation of the objects, the ant-clustering algorithm described previously is applied. The only difference is the calculation of f(i), where the cosine measure (sim(·,·)/α), is used instead of (1 - d(·,·)/α).

After some preliminary tests, it was noticed that the algorithm could never converge to a stable configuration of the grid; that is, the ants were constantly building and destructing clusters. Therefore, one form of modifying ACA in order to promote a stabilization of the grid had to be proposed. The approach adopted here corresponds to gradually cooling down the value of parameter kp so as to reduce the probability of an ant picking up an object after a certain number of iteration steps have passed. With this simple modification, the stopping criterion of the algorithm becomes either a maximum number of cycles (1cycle = 10,000 steps of each ant) or a minimum value for kp. In both cases, the chosen value has to be such that ants are no longer picking up objects from the grid, thus resulting in a final, stable clustering solution.

**CONCLUSIONS**

This paper presented an algorithm for the automatic clustering of text documents using a classification technique inspired by the behaviour of some ant species while organizing and cleaning their nests. As the ant-clustering algorithm used was originally developed to tackle numeric data, some modifications had to be introduced in order to adapt it to deal with text data. Furthermore, we also proposed a cooling schedule for a parameter of the algorithm resulting in the improvement of its convergence properties.

**REFERENCE**

[1] . R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval, Addison Wesley, (1999). | [2] . S. Lawrence, C. L. Giles and K. Bolacker, "Indexing and Retrieval of Scientific Literature", Proc. of the 8th Int. Conf. on Information and Knowledge Management, pp. 139-146, (1999). | [3] S. Lawrence, C. L. Giles and K. Bollacker, "Research Feature Digital Libraries and Autonomous Citation Indexing", IEEE Computer, 32(6), pp.67-71, (1999). | [4] E. Lumer and B. Faieta, "Diversity and Adaptation in Populations of Clustering Ants", Proc. of the 3rd Int. Conf. on Simulation of Adaptive Behaviour: From Animals to Animats, pp. 501-508, (1994). | [5] J. Deneubourg, N. Goss, N. Franks, A. Sendova Franks, C. Detrain and L. Chrétien, "The Dynamics | of Collective Sorting: Robot-Like Ant and Ant-Like Robot", Proc. of the 1st Int. Conf. on Simulation of Adaptive Behaviour: From Animals to Animats, pp.356-365, (1991). | [6] L. N. de Castro, and F. J. Von Zuben, Recent Developments in Biologically Inspired Computing, Idea Group Inc., Hershey-PA, (2004). | [7] E. Bonabeau, M. Dorigo, M. and G. Théraulaz, | Swarm Intelligence from Natural to Artificial | Systems, Oxford University Press, (1999). | [8] J. Kennedy and R. Eberhart, Swarm Intelligence, Morgan Kaufmann Publishers, (2001). | [9] G. Beni, "The Concept of Cellular Robotic Systems", Proc. of the IEEE Int. Symp. on Intelligent Control, pp. 57-62, (1988). | [10] G. Beni, and J. Wang, "Swarm Intelligence", Proc. Of the 7th Annual Meeting of the Robotics Society of Japan, pp. 425-428, (1989). | [11] E. Bonabeau, "From Classical Models of Morphogenesis to Agent-Based Models of Pattern Formation", Artificial Life, 3, pp. 191-211, (1997). | [12] E. Lumer, and B. Faieta, "Exploratory Database Analysis via Self-Organization", Unpublished Manuscript, (1995). | [13] P. Kuntz and D. Snyers, "Emergent Colonization and Graph Partitioning, Proc. of the 3rd Int. Conf. on Simulation of Adaptive Behaviour: From Animals to Animats, pp. 494-500, (1994). | [14] P. Kuntz and D. Snyers, "New Results on an Ant-Based Heuristic for Highlighting the Organization of Large Graphs", Proc. of the IEEE Congress on Evolutionary Computation, pp. 1451-1458, (1999). | [15] 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), pp. 327-351, (1998). | [16] K. Hoe, W. Lai, and T. Tai, "Homogeneous Ants for Web Document Similarity Modeling and Categorization", Proc. of the 3rd Int. Workshop on Ant Algorithms, Lecture Notes in Computer Science 2463, Springer-Verlag, pp. 256-261, (2002). | [18] V. Ramos and J.J. Merelo, "Self-Organized Stigmergic Document Maps: Environments as a Mechanism for Context Learning", Proc. of the 1st Spanish Conference on Evolutionary and BioInspired Algorithms, pp. 284-293, (2002). |