Text Document Classification Using Swarm Intelligence

Mark Jenne and David Pick

Introduction

- Information retrieval systems have contributed significantly to the organization of information
- Success of such systems still depends on maintenance
  - Indexing and cataloging new information
- System for automatic organization of digital documents in PDF format
  - Based on ant-clustering algorithm

Ant Clustering Algorithm

- Proposed in 1994 for analysis and classification of numerical data
- Input data set of N l-dimensional vectors to be clustered
- Input data are spread over a two-dimensional grid of size m x m
- Colony of ants moves on the grid, picking up, carrying, and dropping off objects based on some probabilistic rules

Ant Clustering Algorithm

- Ants have a perception grid of size 3 x 3
- To start the iterative process, objects and ants are placed randomly on the grid
- Ants start moving randomly
  - If not carrying an object and finds an object in its neighborhood
    - picks up with probability inversely proportional to number of similar objects in neighborhood
  - If carrying an object and finds other objects in its neighborhood
    - drops with probability proportional to object's similarity with the perceived ones

Probability Equations

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

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

Standard Ant-Clustering Algorithm

```
procedure ACA (max_it, k_p, k_d)
   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 do,
      for i = 1 to N do,    // for every ant
         if  unladen ant AND cell occupied by item x_i, then
             compute f(x_i) and p_p(x_i)
             pick up item x_i with probability p_p(x_i)
         else if ant carrying item x_i AND cell empty, then
                    compute f(x_i) and p_d(x_i)
                    drop item x_i with probability p_d(x_i)
         end if
         move to a randomly selected neighboring and unoccupied cell
       end for
       t ← t + 1
   end while
   print location of items
end procedure
```

ACA For Text Clustering

- PDFs first converted to text documents
- Transformed into collections of words that will represent an object in the grid
  - Obtained through calculation of relative frequency of a word in the documents

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

ACA For Text Clustering

- Cosine measure is used as a measure of dissimilarity

$$sim(D_D, D_Q) = \frac{\sum_{k=1}^{N} F_{Dk} F_{Qk}}{\sqrt{\sum_{k=1}^{N} F_{Dk}^{\ 2} \sum_{k=1}^{N} F_{Qk}^{\ 2}}}$$

- Each document is transformed into an object which is an l-dimensional vector corresponding to relative frequencies of relevant words
  - Ant-clustering algorithm for text documents is applied

Initial Tests

- Algorithm was not converging to a stable configuration in the grid
- Ants were constantly building and destructing clusters
- To promote stabilization in the grid, picking constant parameter was gradually cooled down
  - reduce probability of an ant picking up an object as more iteration steps pass
- Resulted in a final, stable clustering solution

Performance Evaluation

Table IV – Simulation results for the data set containing 90 documents. *No*: number of objects in the cluster; *Nw*: number of objects classified incorrectly.

Cluster label	*Group*	*No*	*Nw*
C_1	EC	13	1
C_2	FS	9	2
C_3	ANN	3	0
C_4	FS	3	1
C_5	FS	9	3
C_6	ANN	9	4
C_7	ANN	8	2
C_8	FS	19	9
C_9	ANN	11	4