Automatic Generation of Web Portals Using Artificial Ants

Hanene Azzag
Laboratoire d'Informatique
Polytech Tours
64, Avenue Jean Portalis
37200 Tours, France
hanene.azzag@univ-tours.fr

Gilles Venturini
Laboratoire d'Informatique
Polytech Tours
64, Avenue Jean Portalis
37200 Tours, France
venturini@univ-tours.fr

Christiane Guinot
CE.R.I.E.S
20 rue Victor Noir
92521 Neuilly sur Seine
Cédex, France
christiane.guinot@ceries-lab.com

ABSTRACT

We present in this work a new model (named AntTree) based on artificial ants for document hierarchical clustering . This model is inspired from the self-assembly behavior of real ants. We have simulated this behavior to build a hierarchical tree-structured partitioning of a set of documents, according to the similarities between these documents. We have successfully compared our results to those obtained by ascending hierarchical clustering.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Clustering

General Terms: Algorithms.

Keywords: Portals sites, web, artificial ants, hierarchical clustering.

1. INTRODUCTION

The goal of this work is the automatic construction of portal sites for the Web. A portal site can be viewed as a hierarchical partitioning of a set of documents. One of the major problems to solve in this area is the automatic definition of the hierarchy of documents. In actual systems [2, 5, 3, 1], the practice is to have manual maintenance of a document hierarchy, which requires a number of human experts to check each new document, to evaluate the documents and to find the right position in the hierarchy, if one works with an important number of documents then standard approaches are useless.

In our work, we propose a new approach which automatically builds a tree-structured partitioning of the documents. This method simulates a new biological model of ants that is described in [6]: these insects may become fixed to one another to build live structures with different functions. Ants may thus build "chains of ants" in order to fill a gap between two points, or build a nest by closing the edges of a leaf, or form "drops of ants", a function which is not yet well understood. After a given time, chains structures disaggregate (disconnection of ants).

2. THE ARTIFICIAL ANTS ALGORITHM

In our algorithm, each ant $a_i, i \in [1, N]$ represents one document d_i to cluster. An ant a_i is moving over the support or over an other ant denoted by a_{pos} (see the ants colored in gray on figure 1). The similarity between documents (denoted by $\mathrm{Sim}(i, j)$) is based on cosine measure [4] which encodes each text d_i and d_j as a vector of word count. We have used a common weighting scheme, i.e. *tf-idf* (term frequency - inverse document frequency) to represent these vectors. *tf* denotes the word count of the document and *idf* denotes the inverse document frequency (document frequency is the number of documents which contain the considered word). We have also use "Zipf's law" to remove words which occurrence in a documents collection is too high or too low.

The main principles of our deterministic algorithm are the followings: at each step, an ant a_i is selected in a sorted list of ants (data have been sorted according to decreasing order of the average similarity between each others). The first ants on the database will first become connected first in the tree. Therefore the order of the data is important for our algorithms. We simulate an action for a_i according to its position a_{pos}. Let us consider now that ant a_i is located on an ant a_{pos} and that a_i is similar to a_{pos}. As will be seen in the following, when an ant moves toward another one, it means that it is similar enough to that ant. So a_i will become connected to a_{pos} provided that it is dissimilar enough to ants connected to a_{pos}. a_i will thus form a new sub-category of a_{pos} which is as dissimilar as possible from the other existing sub-categories. For this purpose, let us denote by $T_{Dissim}(a_{pos})$ the lowest similarity value which can be observed among the daughters of a_{pos}. a_i is connected to a_{pos} if and only if the connection of a_i decreases further this value. The test that we perform consists in comparing a_i to the most similar ant a^+ (a^+ is ant connected to a_{pos} which is the most similar to a_i). If these two ants are dissimilar enough ($Sim(a_i, a^+) < T_{Dissim}(a_{pos})$), then a_i is connected to a_{pos}, else it is moved toward a^+. Since this minimum value $T_{Dissim}(a_{pos})$ can only be computed with at least two ants, then the two first ants are automatically connected without any test. This may result in "abusive" connections for the second ant. Therefore, the second ant is removed and disconnected as soon as a third ant is connected (for this latter ant, we are certain that the dissimilarity test has been successful). When this second ant is removed, all ants that were connected to it are also dropped, and all these ants are placed back into the support (see figure 1).

When ants are placed back on the support, they may find another place where to connect using the same behavioral rules. It can be observe that, for any node of the tree, the value $T_{Dissim}(a_{pos})$ is only decreasing, which ensures the termination and convergence of the algorithm.

One should notice that this tree will not be strictly equivalent to a dendogram as used in standard hierarchical clustering techniques: each node in our tree will correspond to one data while this is not the case in general for dendrograms, where data only correspond to leaves. Nevertheless, it is possible to interpret our tree in different ways.

WWW 2005, May 10–14, 2005, Chiba, Japan.
ACM 1-59593-051-5/05/0005.

Databases	Size (# of documents)	Size (Mb)	C_r	AntTree $Disc$				AHC			
				Ec	C_f	P_r	T	Ec	C_f	P_r	T
Reuters	1025	4.05	9	0.20	10	0.50	17.78	0.25	3	0.50	47.23
CERIES	258	3.65	17	0.20	10	0.33	0.67	0.29	4	0.26	0.50
Database 1	319	13.2	4	0.18	9	0.77	1.04	0.13	3	0.75	0.84
Database 2	524	20	7	0.03	8	0.92	1.93	0.13	4	0.67	3.39

Table 1: Comparative results obtained between AntTree and AHC.

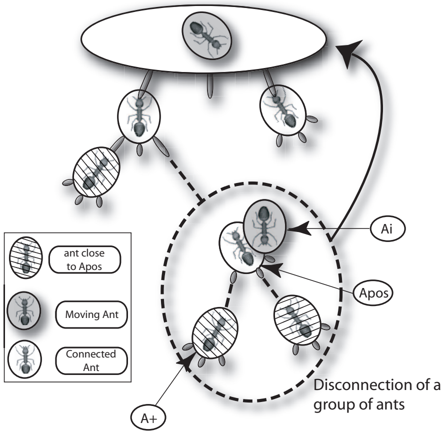


Figure 1: General principles of tree building with artificial ants.

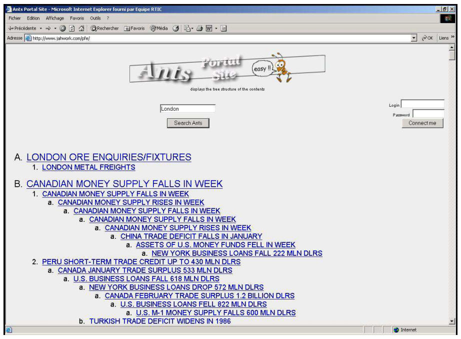
3. RESULTS AND PERSPECTIVES

We have used four databases to study the properties of our algorithm. The *Reuters* databases contains 1025 texts. The *CE.R.I.E.S.* database contains 258 texts dealing with human skin (the *CE.R.I.E.S.* is a research center funded by Chanel). *Database 1* consists of web pages from different scientific topics (73 about scheduling, 84 about pattern recognition, 81 about TcpIp network, 94 about vrml courses). *Database 2* consists of web pages with general topics (55 about c++ courses, 82 about the Danone food company, 86 about IEEE, 90 about cinema, 50 about the Le Monde newspaper, 63 about the Sfr phone company, 101 about medicine information).

Results are presented in table 1 (see a simplified demonstration in http://www.antsearch.univ-tours.fr/webrtic). The real classes of documents (C_r) are of course not given to the algorithms. They are used in the final evaluation of the obtained partitioning. The evaluation of the results is performed with the number of found clusters C_f, with the purity P_r of clusters (percentage of correctly clustered documents in a given cluster), and the classification error measure, denoted by Ec. This measure represents the proportion of document couples which are not correctly clustered, i.e. in the same real cluster but not in the same found cluster, and vice versa.

As far as the number of classes is concerned, AntTree obtains the best results compared to AHC, except for Database 1 where the number of classes is small. The explanation is that AHC often get a lower number of classes than the other algorithms. The automatic cutting procedure of the dendogram considers the largest value of the Ward criterion. The classification error and purity values are comparable (but better for AntTree) for AHC and AntTree, except for Database 2 where AHC does not perform well. Disconnecting the ants and placing them back onto the support seems to increase the performances. Each ant have another chance to change its position and move on others ants perhaps more similar than that on which it is. In our model it is very hard to compute the complexity of the algorithm due to the disconnection of ants but we give the computation time (T) needed once the similarity measure has been computed. The tests were performed on a standard PC (Pentium 2GHz, 512Mo). AntTree algorithms outperform AHC when the databases are large. This is due to the fact that AntTree exploits the tree structure very well and avoids exploring the whole tree when connecting a new ant.

Once all the texts have been clustered in a tree, we encode it in a database in a few seconds and we dynamically generate the corresponding portal site using PHP language (see figure 2).


Ants Portal Site - Microsoft Internet Explorer fourni par Equipe RTIC
Fichier Edition Affichage Favoris Outils ?
Précédente Rechercher Favoris Média
Adresse http://www.jahwork.com/pfe/
OK Liens
Ants Portal Site
easy !!
displays the tree structure of the contents
London
Search Ants
Login
Password
Connect me
Internet

A. LONDON ORE ENQUIRIES/FIXTURES
1. LONDON METAL FREIGHTS

B. CANADIAN MONEY SUPPLY FALLS IN WEEK
1. CANADIAN MONEY SUPPLY FALLS IN WEEK
a. CANADIAN MONEY SUPPLY RISES IN WEEK
a. CANADIAN MONEY SUPPLY FALLS IN WEEK
a. CANADIAN MONEY SUPPLY FALLS IN WEEK
a. CANADIAN MONEY SUPPLY RISES IN WEEK
a. CHINA TRADE DEFICIT FALLS IN JANUARY
a. ASSETS OF U.S. MONEY FUNDS FELL IN WEEK
a. NEW YORK BUSINESS LOANS FALL 222 MLN DLRS
2. PERU SHORT-TERM TRADE CREDIT UP TO 430 MLN DLRS
a. CANADA JANUARY TRADE SURPLUS 533 MLN DLRS
a. U.S. BUSINESS LOANS FALL 618 MLN DLRS
a. NEW YORK BUSINESS LOANS DROP 572 MLN DLRS
a. CANADA FEBRUARY TRADE SURPLUS 1.2 BILLION DLRS
a. U.S. BUSINESS LOANS FELL 822 MLN DLRS
a. U.S. M-1 MONEY SUPPLY FALLS 600 MLN DLRS
b. TURKISH TRADE DEFICIT WIDENS IN 1986

Figure 2: A typical portal site generated from the *Database 2*.

As perspectives, we are currently including the automatic extraction of keywords in order to automatically annotate the subcategories of a given node. An other direction of research is to perform a sampling of the database when it contains many documents (e.g. more than 3000 pages for instance): the texts which have not been used for learning the tree are assigned to a category by following the path with the highest similarity.

4. REFERENCES

[1] D. Filo and J. Yang. Yahoo!, 1997.

[2] R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. On semi-automated web taxonomy construction. In *WebDB*, 2001.

[3] A. K. McCallum, K. Nigam, J. Rennie, and K. Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127–163, 2000.

[4] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. In *information processing and management*, volume 25, pages 513–523, 1988.

[5] M. Sanderson and W. B. Croft. Deriving concept hierarchies from text. In *Research and Development in Information Retrieval*, pages 206–213, 1999.

[6] G. Theraulaz, E. Bonabeau, C. Sauwens, J.-L. Deneubourg, A. Lioni, F. Libert, L. Passera, and R.-V. Solé. Model of droplet formation and dynamics in the argentine ant (linepithema humile mayr). *Bulletin of Mathematical Biology*, 2001.