On Building Graphs of Documents with Artificial Ants

Hanane Azzag
Laboratoire d'Informatique
de l'Université Paris-Nord
99 avenue Jean-Baptiste
Clément
93430 Villetaneuse, France
hanane.azzag@lipn.univ-paris13.fr

Julien Lavergne,
Gilles Venturini
Laboratoire d'Informatique
de l'Université de Tours
64 avenue Jean Portalis
37200 Tours, France
{julien.lavergne,venturini}@univ-tours.fr

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

ABSTRACT

We present an incremental algorithm for building a neighborhood graph from a set of documents. This algorithm is based on a population of artificial agents that imitate the way real ants build structures with self-assembly behaviors. We show that our method outperforms standard algorithms for building such neighborhood graphs (up to 2230 times faster on the tested databases with equal quality) and how the user may interactively explore the graph.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Clustering

General Terms

Algorithms

Keywords

Web, documents, graph, interactive visualization, clustering, artificial ants

1. INTRODUCTION

We deal with the following problem: we consider a set of n documents that may represent the results of a search engine, or the browsing history of a user, or even a set of web pages to be turned into a hypertext by adding hyperlinks between them. We would like to build a graph that highlights the neighborhood relationships between these documents, i.e. a graph where nodes are documents and edges between documents represent the similarity between them. In addition, we would like to display this graph in an interactive way that would allow the user to easily browse the documents.

There are several standard methods for building neighborhood graphs and one may mention for instance the Delaunay triangulation, the Gabriel graphs and the relative neighborhood graphs [7] (RNG in the following). These methods compute graphs from a data set and more precisely from the distances (or similarities) between these data. They can especially be useful for exploring a data set with a content-based approach (i.e. from neighbors to neighbors). However, they are limited by a high complexity ($O(n^3)$ for instance) which restricts their use to small data sets [4].

WWW 2007, May 8–12, 2007, Banff, Alberta, Canada.
ACM 978-1-59593-654-7/07/0005.

2. MAIN ALGORITHM

Our model, called AntGraph, is inspired from the previous algorithm AntTree [1] that learns a hierarchical clustering of documents. It is based on the self-assembly behavior of real ants that progressively build a living structure by connecting their bodies together.

We have generalized this principle for building graphs. We consider one randomly selected ant denoted by a_0. This ant will be the support of the structure and the input node of the graph. Then, the remaining ants are introduced one by one in the graph. Let a_i denotes such an ant. a_i moves in the graph until it finds a convenient location where to connect. For this purpose, a_i follows the path of maximum similarity. Let a_{pos} denotes the ant (node) where a_i is located. The following cases have to be considered: 1) a_{pos} does not have any neighbor to explore: a_i connects to a_{pos}, 2) a neighbor of a_{pos} is more similar to a_i than a_{pos}: a_i moves to the most similar neighbor of a_{pos}, 3) a_{pos} is the most similar ant to a_i (in its neighborhood). a_i connects to a_{pos} and to all neighbors of a_{pos} which similarity to a_i is above a given similarity threshold S_t that is computed as follows: $S_t = \alpha * sim(a_i, a_{pos})$ with $\{\alpha \in \mathbb{R}, 0 \leq \alpha \leq 1\}$. The graph is thus built incrementally. Ants become rapidly connected because they follow the path of greatest similarity, a way to cut through large sets of documents.

3. RESULTS AND INTERACTIVE VISUALIZATION

We have compared our approach with RNG on several sets of web pages. Two documents are connected in the RNG graph if there is not any third document closer to each of them [7]. The same similarity measure is used for both methods: we have used a vector-based representation of these texts and the cosine measure and tfidf weighting scheme [6].

As can be observed in table 1, our method obtains a graph of documents in a time which is significantly lower than RNG. This is one of the most important advantages of AntGraph. Then, we have checked that the graphs we obtained are valid with respect to the similarity measure. For this purpose, we have measured the differences between our graphs and RNG. Due to limited space, we will not give all the tables but we mention our conclusions: AntGraph creates more links than RNG, but those added links represent high similarity values. RNG is known to create a small number of links in general.

Data set	# of documents	Real classes	AntGraph T_{Exec}	RNG T_{Exec}
Webace1 [5]	185	10	0,0011 [0,0001]	0,0983 [0,0099]
CE.R.I.E.S. [3]	258	17	0,0018 [0,0002]	0,2512 [0,0273]
Antsearch [1]	319	4	0,0024 [0,0001]	0,4964 [0,0638]
Webace2 [5]	2340	6	0,0650 [0,0080]	145,2970 [5,4810]

Table 1: Sets of web pages used in our comparative tests and execution times in seconds. Both methods have been programmed in Java and tests were performed on a Pentium-M 2Ghz with 1Go RAM.

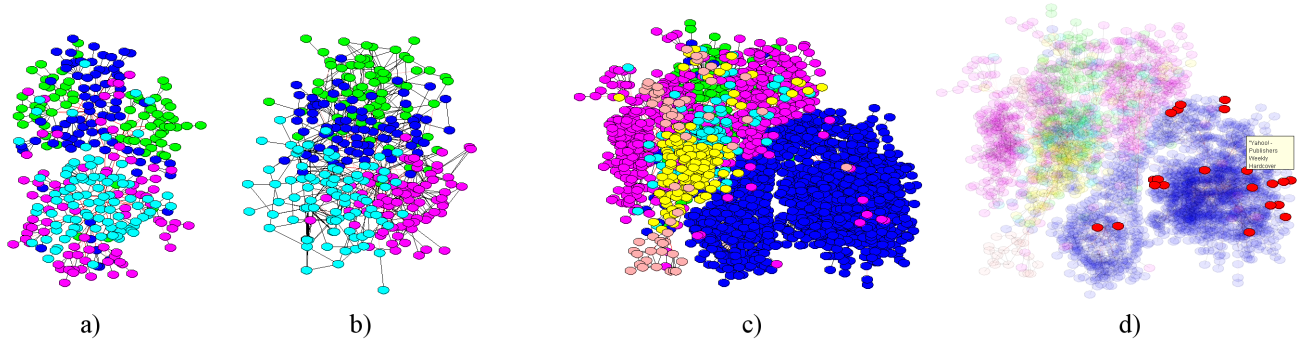

Figure 1: The two graphs on the left are respectively AntGraph (a) and RNG (b) results obtained with the Antsearch dataset (results from Google to 4 different queries). On the right, the Webace2 data set with 2340 documents (c), and the search results for the keyword "weekly" (d).

We have visualized for each method the discovered graphs using forces and springs algorithms [2]. We set a distance between nodes that is related to the similarity between documents. In such graphs (see figure 1), one may easily distinguish the real classes of the documents (which are not given to AntGraph). The constructed graph thus represents the similarities with a high accuracy.

Finally, we have developed a Java-based interface for exploring the graphs in a visual and interactive way. The user may easily view the groups of documents, and which documents are at the center or the frontier of a group, as well as isolated documents. Several interactive actions are possible, like annotating nodes, searching the documents for keywords, zooming with distortion, opening a document, etc. (see figure 1).

4. CONCLUSION AND PERSPECTIVES

We have presented in this paper a new bio-inspired method for building graphs of Web documents from a similarity measure. We have experimentally shown that this method obtains results of high quality and in a very short time compared to a standard method (RNG). Furthermore, we have integrated this construction algorithm in a visual interface that allows the user to explore the set of documents with several interactive possibilities. Among the perspectives that can be derived from this work, we would like to combine our incremental construction with the visualization itself, rather than having two separated phases. In order to deal with larger sets of documents, we want to store several documents within the same node. Finally, we would also like to use thumbnail views of the documents to give an overview of their type and content.

5. REFERENCES

[1] H. Azzag, C. Guinot, and G. Venturini. Anttree: web document clustering using artificial ants. In R. L. de M'antaras and L. Saitta, editors, *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI 04)*, pages 480–484. IOS Press, 8 2004.

[2] T. Fruchterman and E. Reingold. Graph drawing by force-directed placement. In *Software - Practice and Experience*, volume 21, pages 1129–1164, 1991.

[3] C. Guinot, D. J.-M. Malvy, F. Morizot, M. Tenenhaus, J. Latreille, S. Lopez, E. Tschachler, and L. Dubertret. Classification of healthy human facial skin. Textbook of Cosmetic Dermatology Third edition, 2003.

[4] H. Hacid and D. A. Zighed. An effective method for locally neighborhood graphs updating. In *DEXA 2005*, pages 930–939, 2005.

[5] E.-H. Han, D. Boley, M. Gini, R. Gross, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, and J. Moore. Webace: a web agent for document categorization and exploration. In *AGENTS '98: Proceedings of the second international conference on Autonomous agents*, pages 408–415, New York, NY, USA, 1998. ACM Press.

[6] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523, 1988.

[7] G. T. Toussaint. The relative neighborhood graphs in a finite planar set. In *Pattern recognition*, chapter 12, pages 261–268. 1980.