Generating Maps of Web Pages using Cellular Automata

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

David Ratsimba
Laboratoire ERIC
Université de Lyon2
Bat. L, 5 avenue Pierre Mendés-France
69676 Bron Cedex
dratsimb@club-internet.fr

David Da Costa
Polytech'Tours
64, Avenue Jean Portalis
37200 Tours, France
david.dacosta@univ-tours.fr

Gilles Venturini
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
christiane.guinot@ceries-lab.com

1. INTRODUCTION

The aim of web pages visualization is to present in a very informative and interactive way a set of web documents to the user in order to let him or her navigate through these documents. In the web context, this may correspond to several user's tasks: displaying the results of a search engine, or visualizing a graph of pages such as a hypertext or a surf map. In addition to web pages visualization, web pages clustering also greatly improves the amount of information presented to the user by highlighting the similarities between the documents [6]. In this paper we explore the use of a cellular automata (CA) to generate such maps of web pages.

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

General Terms: Algorithms.

Keywords: Web pages, visualization, unsupervised clustering, cellular automata.

2. A CA MODEL FOR DOCUMENTS VISUAL CLUSTERING

In the following, the n documents (or data) to be clustered are denoted by $d_1, ..., d_n$ and $Sim(i,j) \in [0,1]$ denotes the similarity between two documents d_i and d_j. We have considered a 2D CA where the $NCell$ cells are structured on a squared grid.

The set of possible cells states is equal to $S = \{empty, d_1, ..., d_n\}$. In other words, each cell will be either empty or may contain one (and only one) document or data. At each simulation step, the states of cells will be possibly modified according to local transition rules which will aim at letting similar states (documents) appear at close locations on the grid. The size of the grid has been empirically determined as in [3] and is computed with the function $N = E(\sqrt{3n})+1$. This size is supposed to give enough space (N^2 cells but n data only) to the spatial organization of the clusters. The size of the neighbourhood $V(c_{ij})$ is the edge v of the square centered on each cell and is empirically determined by the following formula: $v = E(N/10) + 1$. In our algorithm: a cell is isolated if its immediate $(v = 1)$ neighbourhood contains less than 3 non empty cells. We have decided to obtain non overlapping clusters: thus a state d_i may appear in only one cell at a time. Therefore, we use a list of states denoted by L which represents the list of documents which do not appear on the grid and that remain to be placed. Initially, L contains all the documents and the states of cells are all empty.

Firstly, the local rules for an empty cell C_{ij} are:

- R1 : If C_{ij} is isolated, Then (with probability $1 - P' = 0.25$) $C_{ij}(t+1) \leftarrow d_k$, where d_k is a randomly selected document of L (and provided that $\overline{Sim}_{d'_k \in V(c_{ij})}(d_k, d'_k) > Threshold(t)$)
- R2 : If C_{ij} is not isolated, Then $C_{ij}(t+1) \leftarrow d_k$, where d_k is either a randomly selected document of L (with probability $P = 0.032$), or (with probability $1 - P$) the document of L which is the most similar to C_{ij} neighborhood, (and provided that, in both cases, $\overline{Sim}_{d'_k \in V(c_{ij})}(d_k, d'_k) > Threshold(t)$).

For a cell C_{ij} that contains a document d_k (i.e. $C_{ij}(t) = d_k$), the transition rules are the following:

- R3 : If C_{ij} is isolated, Then $C_{ij}(t+1) \leftarrow empty$ with a probability $P' = 0.75$ (d_k is placed back in L).
- R4 : Else if $\overline{Sim}_{d'_k \in V(c_{ij})}(d_k, d'_k) < Threshold(t)$, Then $Cij(t+1) \leftarrow empty$ and d_k is placed back in L

In all other cases, the cell state remains unchanged $(C_{ij}(t+1) \leftarrow C_{ij}(t))$. The values of P and P' thresholds have been obtained experimentally.

In order to apply these rules to cells and in order to avoid conflicts when assigning on the grid the data of L, we have considered that one cell will evolve at a time (sequential evolution of the CA). A permutation of the N^2 cells is randomly generated at the beginning of the algorithm.

In the rules, we mention a threshold $Threshold(t)$ which evolves over the simulation steps t. This threshold is initialized to the maximum similarity value observed in the data set, and then slowly decreases through the run to the lowest similarity. Initially, the

WWW 2006, May 23–26, 2006, Edinburgh, Scotland.
ACM 1-59593-332-9/06/0005.

Table 1: Results obtained on standard databases. C_F represent the number of found clusters, P_R the purity and E_C the error measure

Databases	Size	Real	Cellular automata			AHC		
	(# of documents)	Classes	E_C	C_F	P_R	E_C	C_F	P_R
AntSearch [4]	332	4	0.35 [0.06]	4.2 [0.6]	0.59 [0.21]	0.17	6,00	0.79
CERIES [1]	259	17	0.62 [0.12]	3.6 [0.9]	0.27 [0.10]	0.36	3,00	0.29
WebAce1 [2]	185	10	0.48 [0.10]	3.8 [0.9]	0.36 [0.21]	0.28	4,00	0.27
WebAce2 [2]	2340	6	0.15 [0.07]	8.0 [0.6]	0.81 [0.27]	0.29	3,00	0.79

documents located close to each others on the grid are thus very similar to each others, thus forming highly similar ”seeds” for the future clusters.

3. RESULTS

We have applied our algorithm on textual databases. In this case, the similarity measure is computed with specific algorithms (*cosinus* and *tf-idf* scheme [5]).

We present in table 1 the obtained results with a comparison with ascending hierarchical clustering. The performances from a clustering point of view are encouraging.

From the grid previously generated by CA algorithm, we have construct a ” browsable ” map: the 2D positions of the documents are respected and the grid is converted into an HTML table. Each cell of the table contains one document and is annotated using the beginning of the document’s title. Then, with JavaScript commands, we may add interactions to the map. Clicking on a cell opens the corresponding document. Zooming the map is possible directly with the browser using the mouse wheel. The resulting map thus represents the similarities between documents, the title of documents and the possibility to zoom and open documents. It is possible to visually evaluate the size of clusters, and also to perform an information retrieval task by exploring the set of documents by their content. Figure 1 present a complete map generated from the Antsearch database. Generating the keywords is very simple (and fast) but gives a basic explanation about the clusters. The beginnings of titles are complementary to each others and provide a good idea of the topic a given area of the map deals with. When one observes the titles, one may notice that these titles have many significant keywords in common. A simple and straightforward extension of this work would consist in extracting the keywords commonly found in every group of 9 cells (the considered cell to be annotated and its 8 neighboors) and to use these keywords for annotation.

4. CONCLUSION

We have presented in this paper a new algorithm for visual clustering which makes use of cellular automata. We have experimentally shown that this algorithm is able to cluster in a relevant way textual databases. The main limitations of our method is the annotation of cells which is very simple, we propose to use a semantic zooming which establishes several hierarchical levels in the map: starting from the initial grid, one may easily group together the cells (by groups of 3×3 cells) and thus make several levels in the visualization. The annotations provided at an upper level could be derived from the previous lower level. This semantic zoom would allow the user to keep a good perception of the global context of the map. Finally, as far as visualization is concerned, we could represent each document using visual attributes that are more informative than a colored cell: one could use for instance thumbnail views of the documents, or other visual attributes indicating the size, type, etc, of documents.

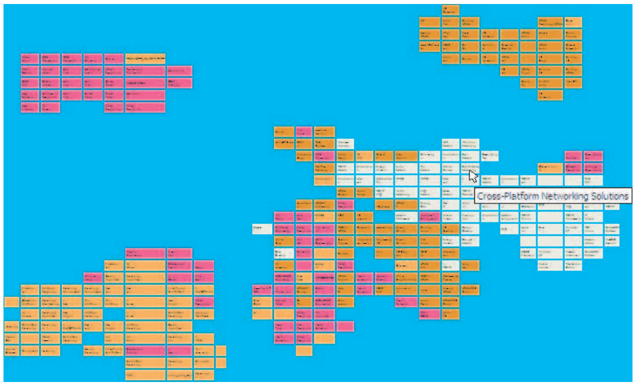

Figure 1: Example of a map generated on the Antsearch databases (319 documents) with annotations

5. REFERENCES

[1] 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.

[2] 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 *Proceedings of the second international conference on Autonomous agents*, pages 408–415. ACM Press, 1998.

[3] E. Lumer and B. Faieta. Diversity and adaptation in populations of clustering ants. In *Proceedings of the Third International Conference on Simulation of Adaptive Behaviour*, pages 501–508, 1994.

[4] F. Picarougne, N. Monmarché, M. Slimane, G. Venturini, and C. Guinot. Two bio-inspired metaheuristics for information search on the web. In *Proceedings of the 6th International Conference on Artificial Evolution*, pages 422–434, Marseille, France, 27-30 octobre 2003.

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

[6] O. Zamir and O. Etzioni. Grouper : a dynamic clustering interface to web search results. In *Computer Networks*, Amsterdam, Netherlands, 1999.