Hy.Doc.: a System to Support the Study of Large Document Collections

Riccardo Rizzo, Giovanni Fulantelli & Mario Allegra

Italian National Research Council-Institute for Educational and Training Technologies, Via Ugo La Malfa 153, 90146 Palermo, Italy

Abstract

In this paper the authors present a system, named Hy.Doc. (Hypertext Document) that allows users to get hypertext-like access to a document collection through automatically created links between individual paragraphs of the documents. The system can be effectively used in educational settings, to support study activities involving large digital information sources such as the Internet and CD-ROMs. Hy.Doc. is based on a Self Organizing Map, a neural network widely used to organize large amounts of data in clusters ordered in a two dimensional array such as a grid or a map. The prototype of the system has been tested by using the documents taken from the proceedings of the AACE WebNet 96 conference.

Key words: Artificial Neural Networks, Hypertext.

Introduction

According to [4] "Hypermedia lets us organize information in accordance with the ways in which we naturally access and manipulate it. This tool presents information in such a way that we can readily see the conceptual associations between information chunks, revealing the structure and interrelationships within the information." Moreover users can browse the information chunks (or atoms) thus following the conceptual associations between them. Therefore, the Hypermedia paradigm is extremely effective in organizing information and letting users gain access to it. However, structuring large sets of documents according to this paradigm is a really complex work: firstly, it is necessary to extrapolate the information nodes from the whole documentation; after that, these nodes must be analyzed one by one in order to identify the associative links among them. As a consequence, hypertext developers should be supported by automatic or semi-automatic processes to handle large sets of documents.

In [2] we have presented a system, based on a Self Organizing Map (SOM network) [1], that supports hypertext authors in identifying the potential links between information chunks. In particular, the SOM network organizes information nodes into clusters and spreads them over a map according to the "semantic distance" between the information nodes; we have proved that the distribution of links in a hypertext is significantly approximated by the "semantic distances" calculated by a SOM network. Therefore, hypertext authors are provided with the map generated by the SOM network, and the links can be searched for between nodes in the same clusters or in clusters close to each other (the authors can visualize the content of a document and the title of the other documents in the same cluster and in the clusters nearest to it).

Note that in [2] the map has been generated starting from information chunks. The same approach can be generalized to get to a hypertext-like organization of sets of documents. In [6], [7], an SOM network has been used to produce, starting from a set of documents, an ordered document map in which a user can navigate and find the right document using explorative search. However, in order to identify hypertext links starting from a collection of documents (like scientific papers or technical reports), a further step is necessary. In fact, documents are not information atoms, they do not carry (or explain) a unique idea or concept, they have an introductory part, they have to explain the fundamental ideas and describe the new ones and finally they have a conclusion. In short they are composed of many information chunks, and it is necessary to break down each document into information atoms.

In Hy.Doc. links are generated between the paragraphs of the documents; each paragraph is believed to concern a single topic and therefore it can be considered as an information atom of a hypertext. As a

consequence, classification of paragraphs is expected to be much more precise than classification of the whole documents. For these reasons, the link structure generated by Hy.Doc. is more coherent and meaningful than the one generated between whole documents.

Finally it should be noted that, since a map of paragraphs can be misleading and difficult to be read for the end user, a map of documents is also developed and visualized in order to support browsing of the collection.

Artificial Neural Networks

Artificial neural network models (ANN) are a dense interconnection of simple non linear computational elements often based on our present understanding of biological nervous systems. These models exhibit some of the features of the biological prototypes such as the capability to learn by example and to generalize beyond the training data. Their ability to learn by example makes the neural networks attractive for complex applications, when it is difficult to write a set of rules or to write a computer algorithm.

Artificial neural networks can be trained to perform a specific task during the so-called "learning stage". In this stage the networks update their architectures, the connection weights or other parameters according to a specific set of inputs (the *training* set). Usually many ANNs are trained at the same time. After the learning stage the performance of the trained networks are compared using the set of *validation* data, and the best network is selected. The network chosen is then tested again using the set of *test* data to evaluate its performance on the real problem. Artificial neural networks can be classified regarding their architecture in recurrent and feed forward networks: in recurrent networks there are connections between the output and the input, on the contrary in feed forward networks the data flow directly from input to output. ANN can also be classified regarding their learning paradigm in supervised learning and unsupervised learning. In supervised learning networks the input and the desired output are both submitted to the network and the network has to "learn" to produce answers as close as possible to the correct answer. In unsupervised learning networks the answer for each input is omitted and the networks have to learn the correlation between input data and to organize inputs into categories from these correlations.

The Self Organizing Map

The Self-Organizing Feature Map (SOM) [1] is an unsupervised neural network, proposed by Kohonen, that during the learning stage tries to build a representation of some features of input vectors. This behavior is typical of some areas of the brain where the placement of neurons is sorted and it often reflects some features of sensorial inputs.

In the SOM, neurons are organized in a lattice, usually one or two dimensional array, that is placed in the input space and is spanned over the input vectors distribution. During the learning stage the neural network creates a cluster of the input vectors.

The number of neurons in the array does not depend on the dimension of the input space, however a small number of neurons may cause a coarse clustering. The SOM algorithm operates a classification where the distance between neurons represents the distance between two clusters of vectors in the input space. Using a two dimensional SOM network it is possible to obtain a map of the input space where closeness between units in the map represents closeness of clusters of input vectors.

The Developed System

According to the idea that links between paragraphs of documents are more significant than the ones between the whole documents, the Hy.Doc. system creates the link structure by a SOM neural network applied to the paragraphs; specifically, this SOM, which is called SOM2, produces clusters of paragraphs (Fig. 1).

However, since browsing between single paragraphs out of their context (the whole document) can be misleading, the links structure between paragraphs is transparent to the user: if the user is reading a specific paragraph, the Hy.Doc. provides him/her the links to all the documents containing paragraphs in the same cluster, rather than to the single paragraphs.

For example, as shown in fig. 1, if the user is looking at an interesting idea described in the paragraph 2 of the document 1, the system will answer proposing the documents 2, 3 and 4 that will contain one or more paragraphs related to the one the user is reading (the paragraph 3 of document 2, the par. 2 of document 3 and the par. 2 of document 4).

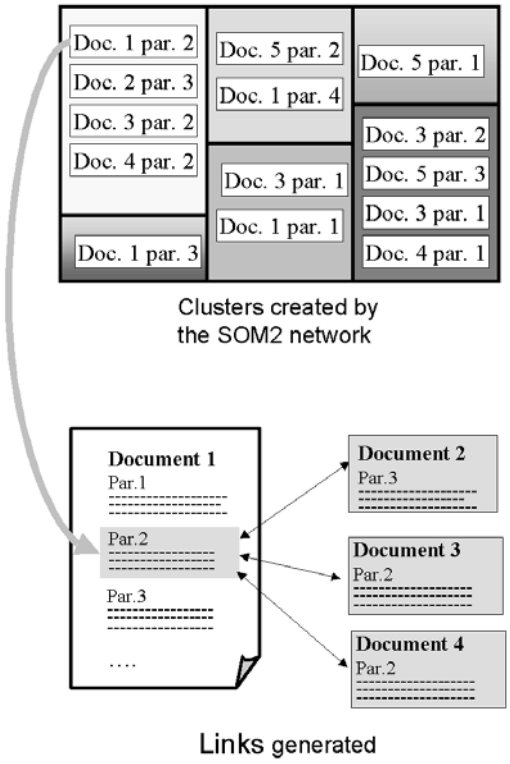

Figure 1: The links between document paragraphs generated by the SOM2 network

In order to make browsing easier for the user, the Hy.Doc. system organizes the collection of documents on a map, through another SOM neural network, called SOM1. In this map the documents are organized in such a way that the ones about the same topic are on the same location on the map or in neighborhood locations. In the Hy.Doc. system this map is represented as a HTML table, as shown in Fig. 2. Each cell on the map in characterized by a set of keywords to help the user understand the subject of the documents contained. The HTML table has been chosen as a visual representation of a bookshelf, that is an effective metaphor for this kind of organization.

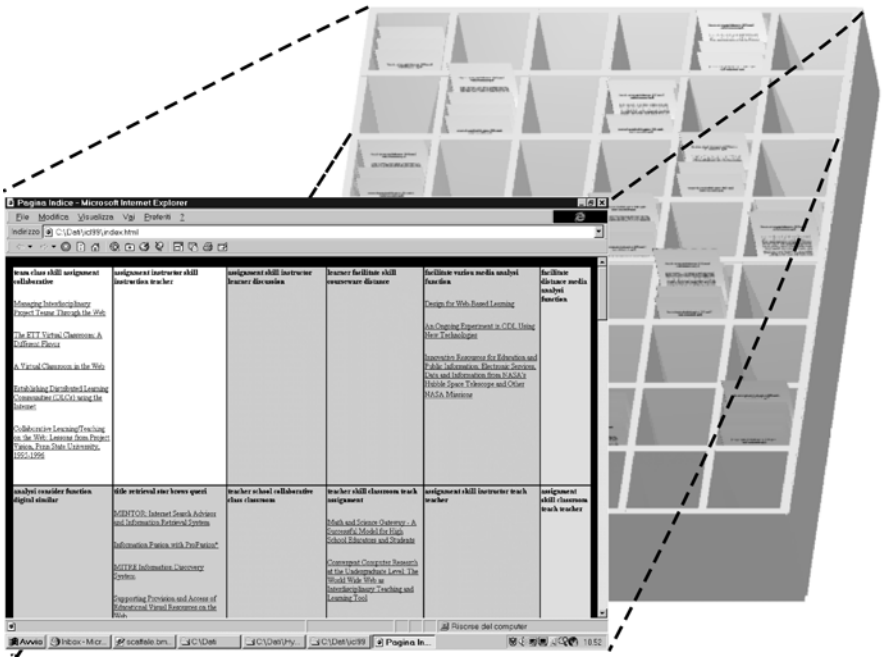

Figure 2: The document organization created by the SOM1 and its representation as a HTML table

Document Representation

The SOM network needs a numerical representation of the document to produce an ordered document map. The vector space information paradigm, a standard practice in information retrieval (IR), has been used in the Hy.Doc. system to encode the document set and to generate the vectors necessary to train the document map. Assuming a dictionary vector D, where each element is a word *di*, each document can be represented as a vector V where the element *vi* is the weight of the word *di* for that document. The word weight can be calculated using the Term Frequency * Inverse Document Frequency (TFIDF) scheme which calculates the "interestingness" value of the word. If the document does not contain this word then *vi* = 0. Assuming a collection of documents B, in the simplest case the weight *vi* of a word *di* in a document T is given by:

$$v_i = tf(i)\log\frac{n}{df(i)}$$

where t*f(i)* is the number of times *di* appears in T (term frequency), *df(i)* is the number of documents in B which contain *di* (document frequency), *n* is the number of documents in B. It is possible to note that the *vi* value will be low if *df(i)* will be near *n*, meaning that the word is common in the document collection. On the converse if a word identifies a subset of documents (i.e. is contained only in few documents) *df(i)* will be low and the *vi* value of the word will be high. In [5] it is possible to find another formula to calculate word weight that is an improvement of the one above, this formula allows the normalization of the vectors used in this work to train the SOM networks to create the information map.

The Prototype of the System

The Hy.Doc. system is based on a set of Java and JavaScript programs and a servlet running with an Apache web server; its first prototype has been used to organize the proceedings of the AACE WebNet 96 Conference that contains 201 scientific papers. These papers have been broken down into 1170 paragraphs by a Java program that tries to recognize the paragraph titles by analyzing the HTML tags. It should be noted that the HTML is a mark-up language and does not allow users to specify the structure of the documents; as a consequence, the start of each paragraph is not explicitly indicated, but it has to be identified starting from the document appearance. For example, in order to highlight the heading of a paragraph, the author could use the <B> tag (with a couple of <P> tags to add blank lines) rather than using the <H> tags. Consequently, the paragraph separation is not perfect, but these problems can be avoided by using a fixed document template or by including structural information in the documents, as when using XML.

The representation of the paragraphs has been obtained through the TFIDF technique with a vocabulary of 150 words, so that the SOM2 neural network has been trained by using a set of 1170 vectors of 150 dimensions. Similarly, the same representation technique has been adopted to model the whole documents and the document map has been produced starting from the same vocabulary, in such a way that the training set for the SOM1 neural network is made up of 201 vectors of 150 dimensions. After the learning stage of the SOM2 network, the link structure between paragraphs has been written in a text file. The document map, produced after the learning stage of the SOM1 network, has been finally translated into the HTML table. The file containing the link structure and the HTML table are used by the Java servlet to handle the user interaction.

A user can access the system through a common Internet browser. When a user gets access to the system, the document map is sent by the server and visualized in the user browser. S/He can locate the area of interest on the map, choose a document and visualize it by "pointing and clicking" on the map (when a document is visualized, only its location is shown on the map). Afterwards, the user can select a paragraph of interest from the document and ask the Hy.Doc. system for the other documents that contain paragraphs related to it (Fig. 3a), which are then visualized on the map (Fig. 3b). The user can look at the topic areas of the returned documents and decide whether they are of interest for him/her or not; the abstract of the documents can be requested in order to support this decision.

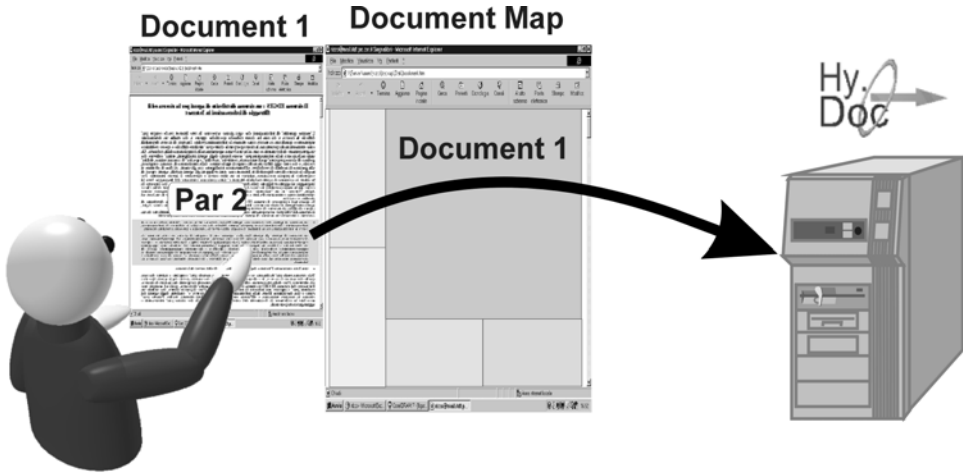

Figure 3a : The user requests the documents connected to the par. 2 of the document 1

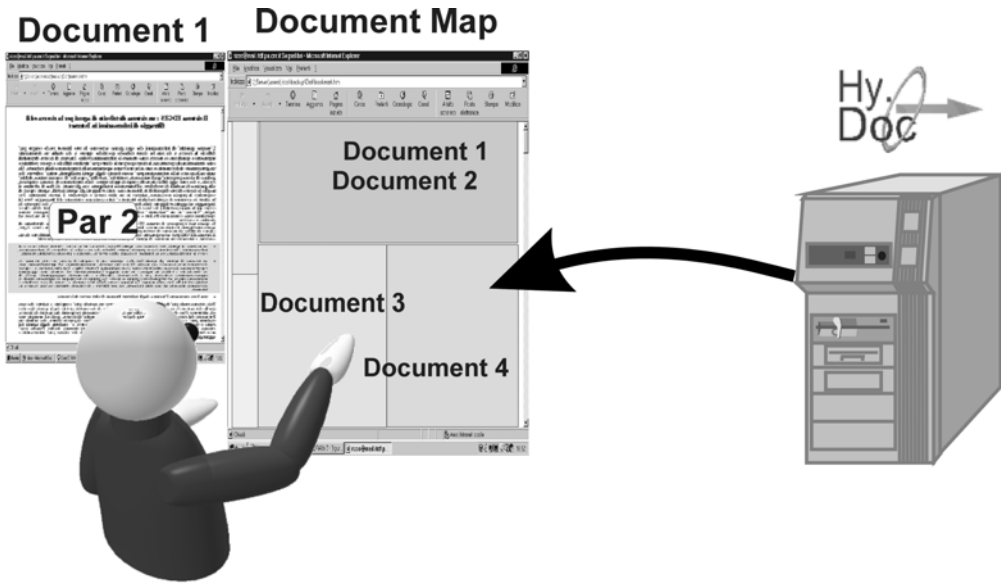

Figure 3b : The Hy.Doc. system returns the related documents

Conclusions and Future Works

Although we do not have the results on an extensive use of the system usage, Hy.Doc. promises to be a useful tool to browse large document collections. There are still many open issues; one of them is the system interface that at the moment visualizes the document map SOM1 as an HTML table where the opportune links to the documents are placed. When a large part of all the 201 documents in the document set are visualized the HTML table becomes too large and makes it impossible to have a good idea of the whole document set. This is one of the major drawbacks of the system because it affects its usability. Another key issue in the interface problem is personalization that is absent in the present prototype. Personalization is very important because it allows users to tailor the system to their needs.

Currently the interface is under development and we expect to add new features soon.

Acknowledgement

The authors want to thank the AACE organization for allowing them to use the proceedings of the WebNet 96 conference.

References

1. Kohonen T.: Self Organizing Maps, Springer-Verlag, Berlin, 1995.

2. Rizzo R., Allegra M., and Fulantelli G., Hypertext-like Structures through a SOM Network, ACM Hypertext' 99.
3. Halasz F.G. : Reflections on Notecards: Seven Issues for the Next Generation of Hypermedia Systems. Communications of ACM, july 1988, vol. 31, number 7.
4. Ginige A., Lowe D. B., Robertson J., : Hypermedia Authoring, IEEE Multimedia, Winter 1995.
5. Balabanovic M., Shoham Y.(1995) "Learning Information Retrieval Agents: Experiments with Automated Web Browsing", Proceedings of the AAAI Spring Symposium on Information Gathering from Heterogenous, Distributed Resources, Stanford, CA, March 1995.
6. Kaski S., Honkela T., Lagus K., Kohonen T., Creating an order in digital libraries with self-organizing maps, in Proc. of WCNN'96, World Congress on Neural Networks, (San Diego, September 15-18, 1996), pp. 814-817.
7. Kaski, S., Lagus, K., Honkela, T., and Kohonen, T., Statistical aspects of the WEBSOM system in organizing document collections. Computing Science and Statistics, 29:281-290. (Scott, D. W., ed.), 1998, Interface Foundation of North America, Inc.: Fairfax Station, VA.