Document Storage and Retrieval in a Neural Database

P. Parodi[1,2], R. Brancaleon[1] , F. Venuti[1], G. Musso[1], V. Torre[1,2]
[1]ISAS, Via Beirut 2-4, 34014 Trieste, Italy; [2]INFM, Trieste, Italy

Abstract

A database of the leech nervous system has been developed. The end-users of the database will mainly be neuroscientists, especially those studying the invertebrate nervous system. It is mainly a document database, collecting papers on the leech nervous system, which is maintained in a largely automatic fashion. The database is composed of three subsystems: (1) an object-oriented, relational database management (sub)system, devoted to storing and maintaining the data and to provide input/ouput tools to the end-users; (2) a document understanding subsystem, which transforms paper documents into an electronic format which can be stored into the database; (3) an information retrieval subsystem, through which end-users extract information from the database. The actual system has been developed as a combination of original modules (the document segmentation module, the field extraction module, parts of the information retrieval subsystem, the user interfaces) with commercial products (the database management subsystem, the OCR module), and a relevant part of the effort has been put into the integration of the several modules into a common framework.

1. Introduction

There is a growing interest in bioinformatics, the combination of biology and informatic tools [2]. Biological databases are becoming essential tools for bioscientists: as evidence of this, it should be remembered that molecular genetics is being transformed by the availability of biological databases. The most famous example of this is given by the Human Genome project, started in 1990.

This paper describes a neural database which is essentially a bibliographic database which collects papers on the leech nervous system. When it will be fed completely, it will have about 1000 documents – therefore, a very small database. A key characteristic of the database is that papers are input to the database by an automatic document understanding process which goes from the article in hardcopy format down to the electronic format of the same article including dealing with images and layout.

Beyond the interest that the database might have to neuroscientists, the endeavor is relevant for many applications: in fact, problems to be solved are similar to those emerging in other situations such as for judiciary, publishers, medical, industrial archives. There is indeed a ubiquitous need for a high degree of automation regarding data integration and searching, and for moving towards more and more sophisticated searching capabilities.

Other examples of neural databases include Flybrain, a database on the fruit fly (drosophila) nervous system. [1], and APLYSIA, a database in construction of the neurons of the aplysia, and many others (for a list of neural databases on line, see http://www.neurogu¡de.com/neuroresac_4.htm¬). Most of neural databases currently available, however, are atlases of neurons with comments inserted manually by the maintainers.

2. An overview of the system

The system is structured as shown in Fig. 1. Roughly, it is composed of three subsystems: the database management system, the acquisition subsystem and the information retrieval subsystem. The database management subsystem is that part of the system which is devoted to storing and maintaining the database, and to providing the user with the basic input/output tools for accessing the database (see for example [4]). It will be briefly

discussed in Section 3. The acquisition subsystem is that part of the system which makes it possible to insert new documents into the database. It will be discussed in Section 4. The information retrieval subsystem is that part of the system by which users are able to query the database and extract information from the documents stored there. It will be discussed in Section 5.

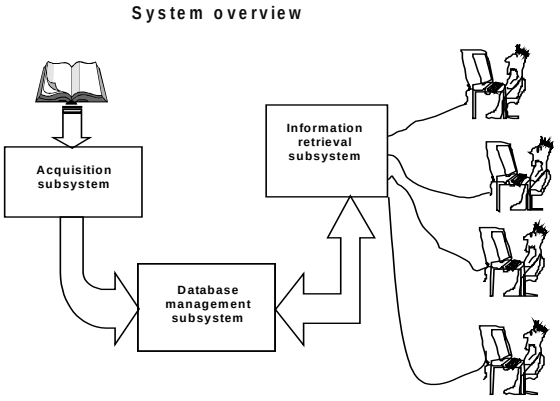


Fig.1. An overview of the system.

3. The database management subsystem

We have used an object-oriented, relational database management system. To ensure that it be standard enough, we have used a commercial product. Among the many advantages offered by a database management system of this sort with respect to a simple file server, we cite the possibility of having many concurrent DB users in network and of having tasks subdivided in a client-server environment (no large amount of computation required on the part of the user).

4. The acquisition subsystem

The acquisition subsystem allows database maintainers to add new documents to the database. More specifically, the document goes through the stages illustrated in Fig. 2. The document is originally in paper format. A scanning device transforms the document into an image in BMP (bitmap) format (Section 4.1). Then the document understanding process, composed by a document segmentation module (Section 4.2), a text extraction module (Section 4.3) and a field-extraction module (Section 4.4), transforms the image into a formatted file which includes text (in a readable format) and pictures and whose layout has been more or less fully interpreted.

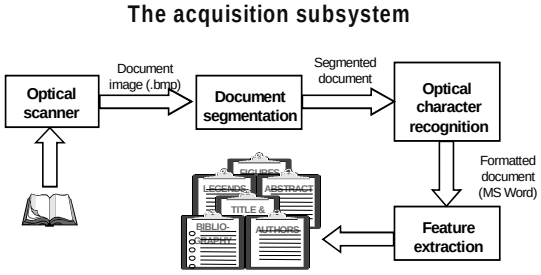


Fig.2. The acquisition subsystem. Paper documents are scanned, translated into digital format (MS Word) and relevant fields (references, title, authors…) are extracted.

4.1 The document-scanning module

The document-scanning module is made of a commercial optical scanner. Documents are scanned as binary images at a resolution of 300 dpi (a standard resolution for document understanding systems).

4.2 The document segmentation module

The purpose of this module is to subdivide each document page into its meaningful parts (e.g. text, images, graphs…) so as to provide a pre-processing for the document interpretation stage, which includes reading text and possibly processing images. It is worth noticing that the document segmentation stage is the critical step in the document recognition process: e.g., the OCR module is generally very reliable once the exact location of text has been determined.

A thorough survey on papers on document segmentation can be found in [6]. In our system we have used an algorithm based on the analysis of background, which has partially been presented elsewhere [7]. This method is accurate and fast, and it deals successfully with documents with an arbitrary layout, documents where graphical features

2

and text are intertwined, skewed documents and noisy documents.

4.3 The text recognition module

The text recognition module is made of a commercial OCR, whose input is the output of the document segmentation module. Beyond reading the textual content of the regions classified as text by the document segmentation module, the OCR module identifies the main parameters of the text (font style, interline spacing...) and the layout of the page, and puts everything into MS Word format.

4.4 The field-extraction module

The output of the previous step is a document page with regions of text and regions of non-text, with its correctly reconstructed layout. The following phase is to identify the main fields of a document:

- title, authors, address…;
- the abstract;
- the section titles;
- the references;
- the figure legends.

This is done by exploiting the fact that there are distinguishing typographical characteristics (font properties, location in the page, special identifiers…) that allow readers quickly to identify each of the fields. These fields are extracted automatically, but the process is supervised by the maintainer which can correct the choices of the algorithm.

Each journal has its standards to identify each of the fields. These standards are stored into the database, so that each time a new paper is inserted into the database, its important fields can be found by comparison with the properties of the different models. When no model is present, a new model is created (see Fig. 3) in a computer-assisted fashion: the maintainer selects portions of the paper and declares they correspond to a particular field. The system then extracts the information on the selected text and it records it as a property of that field for that journal.

5. The information retrieval subsystem

The information retrieval subsystem is for addressing the queries of end-users. Examples of queries are:

- retrieve documents by bibliographic data (e.g., documents cited by a given paper...);
- retrieve documents by topic (e.g., documents concerning post-crash regeneration of synapses);
- retrieve documents by similarity (e.g., documents similar to a given document);
- retrieve documents by (a pre-defined) category (e.g., documents belonging to the «single neuron studies» category).

Except for the retrieval of documents by bibliographic data, which is straightforward, the retrieval method in all of these cases is based on the classical vector-based model (see for example [8,5]). In this model documents, queries and categories are represented by feature vectors. The various types of search are then reduced to comparisons between the feature vectors with an appropriate metric.

6. System integration

The database of the leech nervous system has been developed as a combination of many modules. For the things we had developed an expertise, such as document understanding, we have contributed with original modules (document segmentation, field extraction...). There were also things that we had planned to tackle ex-novo, such as information retrieval, and in this case we have made ad hoc modification to existing commercial software in order to fit our purposes. For the things that were already commercially mature (the database management system, the OCR system), we have bought commercial products.

A relevant effort has been devoted to system integration, and is still being devoted, in order to increase the degree of automation in the acquisition process. Although the different parts of the system rely on software based on different languages (C, C++, SQL, Visual Basic), the overall process is controlled by software written in Visual Basic.

7. Human interfaces

7.1 The acquisition interface

The acquisition interface guides the maintainer, i.e. who feeds the database with new documents, through the insertion process. Fig. 3 gives the scheme according to which the interface works. Upon putting a new document into the database, the maintainer is first asked to provide information on the type of document (book, journal paper…).

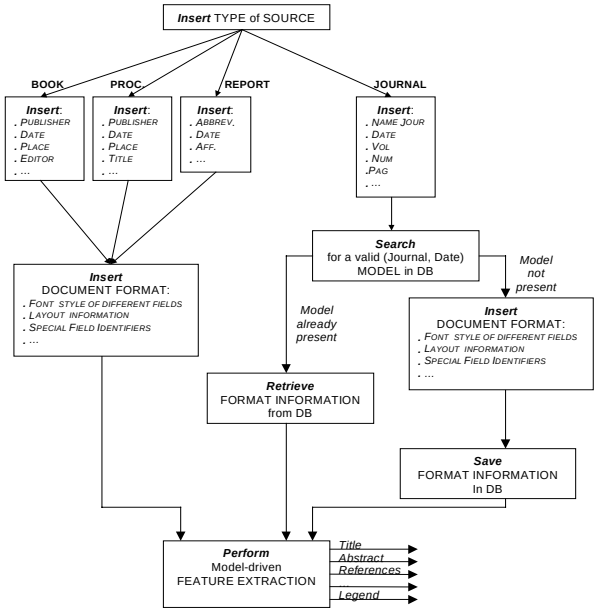


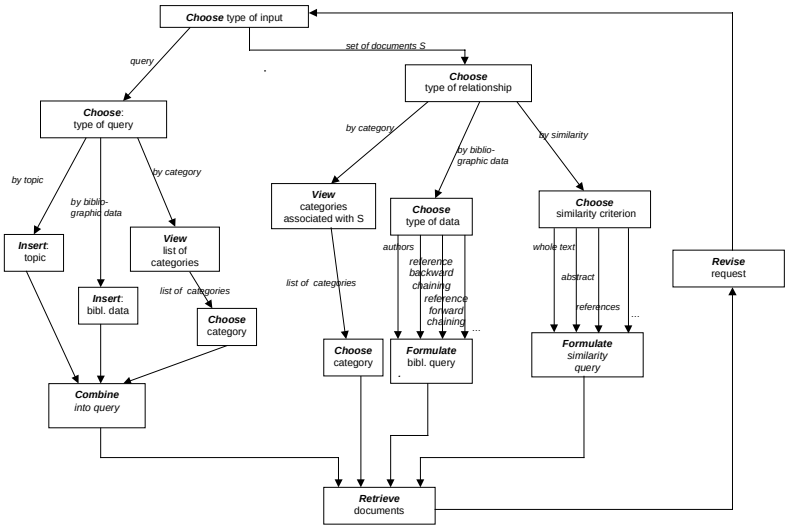
Fig. 3. The working scheme of the acquisition interface.

The maintainer is then asked to provide information on the document formatting (e.g., the reference style, the title position, the font style of the different parts of the paper, etc.) either directly by selecting the corresponding portions of the document or indirectly by giving the journal title and date, so as to access the database of journal styles, which includes the set of features for all journals in all their periods of publication. At this point the document understanding algorithm reads the document and transforms it into a digital version of the document, and the maintainer is able to supervise the process, check whether there have been interpretation problems and possibly correct them.

7.2 The query interface

The query interface guides the user through the information retrieval process. Fig. 4 gives the scheme according to which the interface works. The query input can be either a phrase specified by the user (left branch of the diagram) or a set of documents – possibly a single one – (right branch). An explicit query can be limited to a certain category, or to certain bibliographic data (e.g. a journal, or an author), or even to the set of documents which have been recovered that far. As to the right branch, suppose for simplicity that the set of documents contains a single document. Users can choose to search for similar documents, where «similarity» may be judged by the whole text, the abstract, the set of references, etc. or a combination of these fields. Users can choose to search by bibliographic data, in which case they may search, e.g., for papers cited by a given paper (backward chaining) or for papers citing a given paper (forward chaining). Finally, they can view categories associated to one document and explore them

Fig. 4. The working scheme of the information retrieval interface



further. At the end of each query users can refine it or go back to the initial query.

Fig. 5 provides a view of the interface. The idea is to keep track, as in [3], of all the queries formulated up to a certain point. Furthermore, many queries can be formulated by user-friendly mechanisms. E.g., given a document one can obtain all documents which are similar to it by dragging the icon of the document and dropping it onto the toolbar where the «similar documents» button is located. Similarly for viewing bibliographic data or categories of one or more documents. The icon «?» is to formulate explicit queries that range over the current set of retrieved papers (if not specified otherwise). The shopping cart collects interesting documents during the search.

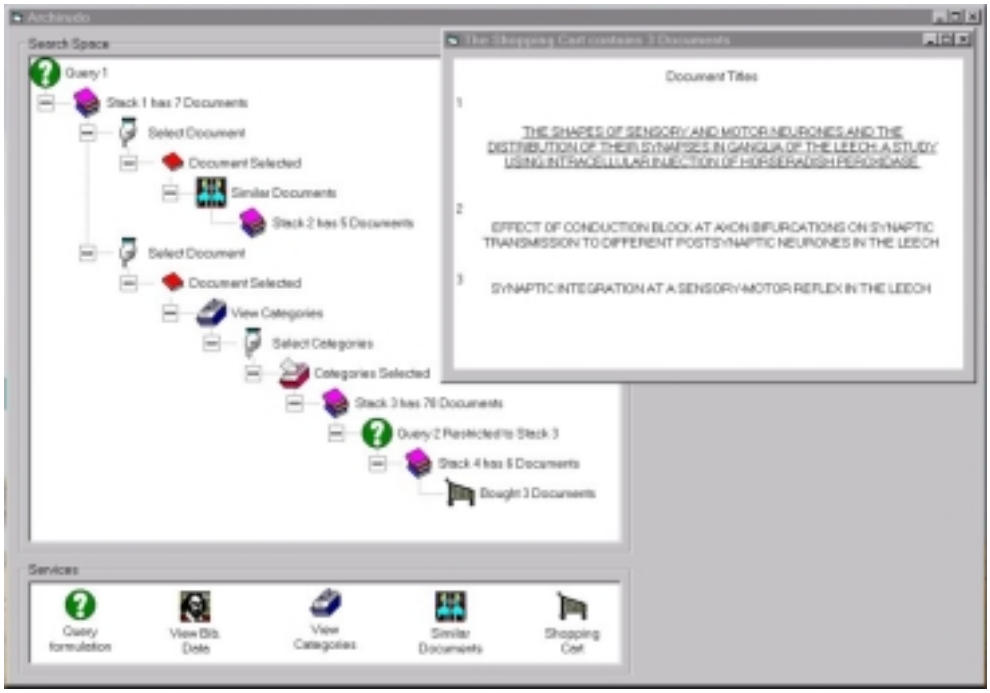


Fig. 5. A view of the information retrieval interface.

References

[1] Armstrong, J.D., Kaiser, K., Müller, A., Fischbach, K-F., Merchant, N. and Strausfeld, N.J. (1995) Flybrain, an on-line atlas and database for the Drosophila nervous system. Neuron 15(1):17-20

[2] Baxevanis, A. and Ouellette, B.F.F. Bioionformatics: A practical guide to the analysis of genes and proteins, John Wiley and sons, 1998

[3] Campbell, I., Supporting information needs by ostensive definition in an adaptive information space, In: Ian Ruthven Ed., MIRO '95, Workshops in computing, Springer Verlag, 1995

[4] Frost, R. A., Introduction to knowledge-base systems, William Collins , sons & Co. Ltd, 1986

[5] R. R. Korfhage, Information storage and retrieval, Wiley Computer Publishing, 1997.

[6] L. O'Gorman and R. Kasturi. Document Image Analysis. MEE Computer Society, 1995.

[7] P. Parodi and G. Piccioli, A fast and flexible statistical method for text extraction in document pages, Proc. of the IEEE Conference of Computer Vision and Pattern Recognition (CVPR), San Francisco, 1996

[8] G. Salton and M. J. McGill, McGraw Hill, Introduction to modern information retrieval, Computer Science Series, 1983