Linguistic-based toolbox for hypertext automatic linking on large technical documentation

Cécile Gros, Didier Bourigault, Jean-Louis Vuldy
Électricité de France
Direction des Études et Recherches
1, avenue du Général de Gaulle, 92141 Clamart Cedex – France
september 94

Introduction

We have developed a terminology extraction software that parses texts on any technical subject and supplies a web of likely terminological units. This web, together with the original corpus it has been extracted from, is organized as an hypertext, known as terminological hypertext, which is submitted to a terminologist for a validation. The original purpose of that hypertext was to allow the terminologist to see terms in their different contexts, but we found it very useful to navigate within the documentation. So, in addition to our previous motivation (automatic terminology extraction), we have started to think about building an hypertext consultation system for large technical documentation. In our firm, this area is becoming more and more accurate as much as technical documentations are avalaible on electronic supports. Our objectives are twofold: first, to develop efficient accesses to information by imagining new types of links in an hypertext system; there must be more to implement than just table of contents and terminological index. Secondly, to provide a linguistic-based toolbox to semi-automatically build these links; we think that linguistic analysis of large corpora can be very useful to automatically skim a large amount of text, even if a manual validation is most of the time necessary to achieve the work.

The structure of the paper is following the progress we have just described. The first section gives an overview of our terminology extraction software, LEXTER. Then, we briefly describe the aims and principles of an hypertext consultation system for large technical documentation and explain our twofold project: implementing new types of links in an hypertext system and providing linguistic-based tools for automatic building.

LEXTER, a terminology extraction software

A presentation

LEXTER is a Terminology Extraction Software (Bourigault, 1992, 1993, 1994). A corpus of French texts on any technical subject is fed in. LEXTER performs a syntactical analysis of this corpus and yields a network of noun phrases which are likely terminological units, representing the concepts of the subject field. This network of likely terms, together with the corpus it has been extracted from, is then passed on to an expert for a validation by the means of a terminological hypertext web.

LEXTER has been developed in an industrial context, wich is the Research and Development Division of Electricité de France, the French Electricity Board. Thus, from the beginning of the project, we had decided to focus upon a strongly restrictive criterium: to apply and test the system over a wide range of texts. The texts analyzed are unrestricted texts gathered in large corpora. We had then to choose a fast and wellproved method. Moreover, we argued that, given the restricted grammatical forms of complex terms, it was not necessary to go into a complete syntactic analysis of sentences to extract likely terms from a corpus. The system uses local syntactic parsing techniques, wich are based on surface patterns, and satisfy the required robustness and accuracy constraints and, at the same time, exempt from a total parsing of sentences. LEXTER software does not complain about the morphological form of words. The first step consists in categorizing all the words (noun, adjective, verb, ...) and replace them by the form they have in a dictionnary.

A complete description of how LEXTER is working is beyond the scope of this paper, it can be found in (Bourigault, 1992, 1993, 1994).

The Terminological Hypertext Web

In an entirely automatic way, LEXTER analyses a whole corpus in batch mode. It is only at the end of this process that the user consults and validates likely terms. LEXTER organizes the terminological web, together with the corpus it has been extracted from, as an hypertextual structure. The interface, that allows to consult the results provided by LEXTER, is an hypertext known as terminological hypertext. This terminological hypertext is composed of nodes and links. The two main types of nodes are term nodes and text nodes.

TERM-nodes Each term-node is associated with a likely term extracted by LEXTER from the original corpus. There are as many term-nodes as likely terms. To each term-node corresponds a display wich provides a large set of informations concerning the term. Figure1 gives a simplified view of the display. The informations provided are :

- The decomposition of the term into a head and an expansion, only if the term is a complex one.
- Numerical coefficients : the number of texts in which the term has been detected, the total number of occurences of the term, composed of the number of occurences of the only term and the number of occurences of the term within more complex terms, the productivity of the term, that is the number of different complex terms of which the term is a part (as a head or an expansion). This coefficient is divided into the productivity of the two grammatical positions (H-position and E-position).
- The list of the identifiers of the texts in which the term has been detected. The division of the corpus into textual units has been made before LEXTER analyses it. A unique identifier has been attributed to each textual unit to allow the system to indicate where the likely terms it proposes come from.
- Two lists of complex terms : those in wich the term is in H-position and those in which it is in E-position.

TEXT-nodes Each text-node is associated with a textual unit of the original corpus. To each text-node corresponds a display in wich, both the textual unit itself and all the terms LEXTER found in it, are displayed.

The types of links are so numerous that we cannot list all of them. Nevertheless, to get some idea of them, one can go back to the descriptions of text-nodes and term-nodes and imagine that a single clicking either on a text identifier or on a likely term allows the user to visualize either the text-node or the term-node. A few examples of hypertextual links are given on figure 1. In order to validate the list of likely terms, the user makes the choice either to retain or to eliminate the terms. He can also add some terms which would not have been detected by the system. At last he can take the initiative for building of some hypertextual links between nodes of his choice.

Some examples of treated documents

As we already pointed it out, LEXTER is domain-independant and can be used on different kinds of technical documents. We have used it to extract terminology from the following corpora :

- Power plant maintenance manuals (200 000 words). They describe the working of several sub-systems in nuclear power plant.
- Electrical networks planners guide (150 000 words). This guide is the technical and administrative reference for people (planners) in charge of anticipating future developments of the electrical networks in France.
- Research projects (1 200 000 words). This corpus consists in approximatly 1700 short texts (one or two pages) in which scientific officers of the Research and Development Division of Electricité de France describe their medium term research projects.
- Technical documentation of the ASTER software (700 000 words). ASTER is a project at EdF to develop a general software for mechanics.

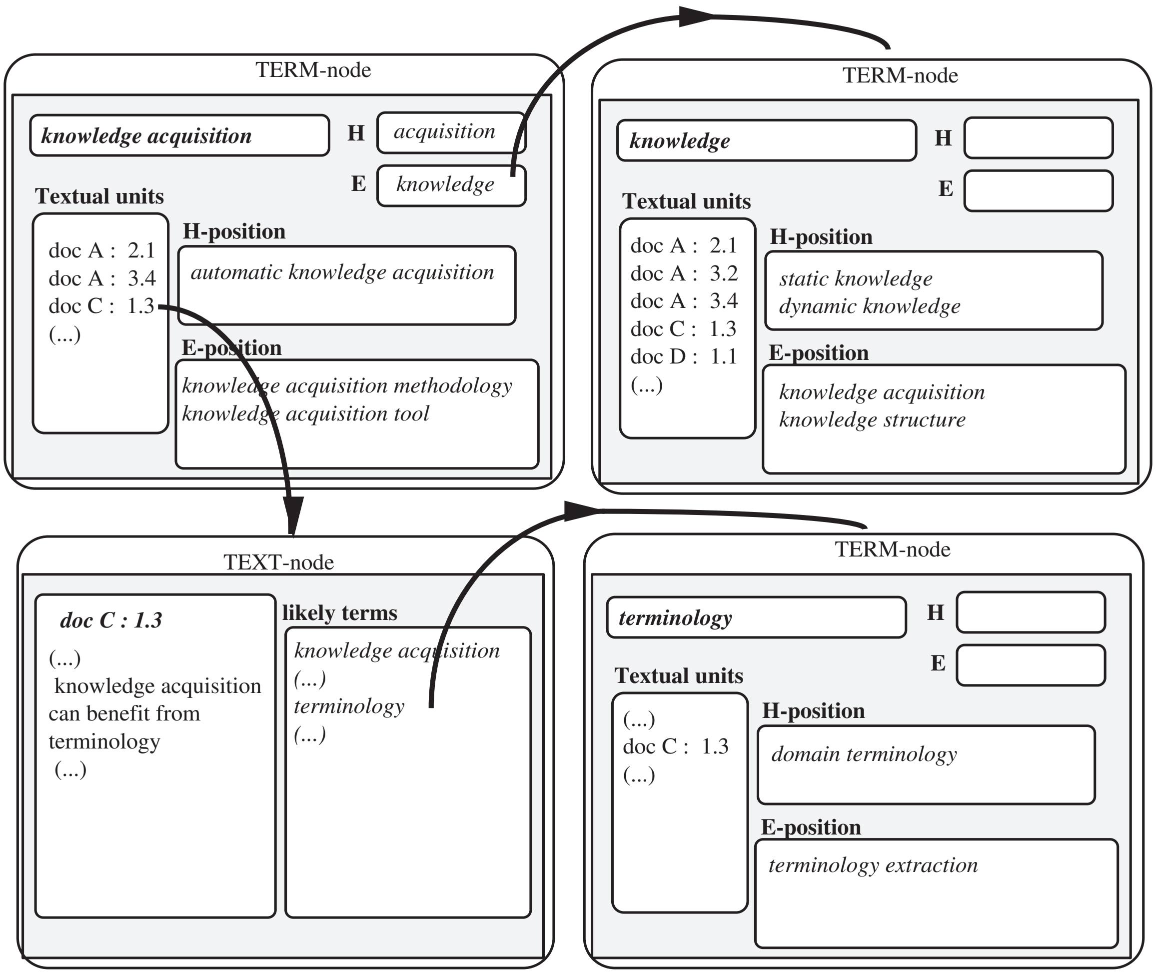

Figure 1: Some hypertext links in the Terminological Hypertext Web.

Hypertext system for large technical documentation

Motivation

Thanks to the development of OCR techniques and mass storage technology, technical documentations are now available on electronic supports and are machine readable. But the exploitation of such an information stays a problem for users with no efficient way to access it. There are two major approaches for helping a user dealing with large text-based information (Jacob and Rau, 1988): first, the user make a query to the system. The answer waited can be either a set of documents in decreasing order of pertinence, or an information extracted from texts. The query can be more or less constrained, ranging from a boolean combination of key words up to a sentence in natural language. The second class of systems lets the user manage his search : there is no direct query to the system, but tools and helps are providing to allow the user to conveniently browse through a collection of documents. In this second class, hypertext systems present an easy and efficient way for structuring and consulting a large set of texts. The nodes of the hyperdocument contain informations and are related by links which set up a non linear collection of texts.

To improve hypertext systems, the major need identified (see for example the proceedings of Hypertext93, Seattle, November 1993) is to introduce more semantics in the components of hypertext webs, in order to find new consultation methods. Linguistic analysis is a way to establish dynamic links between textual units in a hypertext.

Accessing texts

LEXTER allows us to build a structured index for our documentation. We also have the classical table of contents access to the text. These two accesses are very useful and are sufficient in some applications. Nevertheless we want to go further. So we need to develop other accesses to the text, with the objective to provide a linguistic-based toolbox to partially automatize the building of these accesses. According to our experiment with terminology, we do not think that a complete automatic building is possible: we can take advantage of the power of exhaustivity of machine analysis to extract and organize information, but a human validation is still necessary.

Up to now, we have been thinking to two types of links to add to our web : an index of actions and a domain representation made accessible to the user. For the first one, the goal is to detect actions in the text by working both on noun phrases (already extracted by LEXTER) and on verbs. Relations between these actions (such as causality or necessity) can also be detected. We suppose that some linguistic hints could caracterize these actions and relations between them. The second access (domain representation) consists in presenting to users a vision of the technical domain covered by the texts, for example by means of a semantic network, which nodes are related to some documents or part of texts. This idea is not completely new, but the problem is to build the representation without doing all the work by hand. Some people (Ogonowzki and al, 1994) are already working on a toolbox to extract and structure knowledge from large corpora of technical texts in order to construct resources for natural language processing systems. Although our goal is quite different, we think that similar techniques could be useful.

Conclusion

The terminology extraction software LEXTER is a first step in our approach to build an hypertext system for large technical documentation. It provides an hypertext tool to validate the terms automatically extracted from texts by a linguistic analysis. When the validation is achieved, the web of terms is used as an index for the consultation of the documents. We now want to generalize this approach : to extract useful information from stored text by a linguistic analysis, allow a user to validate and organize what has been automatically detected, and use the result to implement hypertext links for consultation. Our work to find new types of links (more than just table of contents and terminological index) for an hypertext consultation system is just beginning, but some promising tracks already appear : index of actions, domain representation accessible to the user.

Some references

Bourigault D. (1992). Surface Grammatical Analysis for the Extraction of Terminological Noun Phrases. In *Proc. 14th International Conference on Computational Linguistics*, Nantes, pp.977–981.

Bourigault D. (1994). LEXTER, a Terminology Extraction Software. Application to Knowledge Acquisition from Texts. Ph.D. Thesis, Ecole des Hautes Etudes en Sciences Sociales, Paris.

Jacob P.S, Rau L.F. (1988). Natural Language Techniques for Intelligent Information Retrieval. In *Proc. of the 11th ACM SIGIR Conference on Research and Development in Information Retrieval*, Grenoble.

Ogonowski A, Herviou M.L, Dauphin E. (1994). Tools for Extracting and Structuring Knowledge from Texts. In *Proc. 15h International Conference on Computational Linguistics*, Kyoto, pp.1049–1053.

Vuldy J.L. (1994). Automatic Generation of Electronic Books. Poster at *ACM European Conference on Hypermedia Technology, ECHT'94*, Edinburg.