TExtractor: a multilingual terminology extraction tool

Antonio S. Valderrábanos
SchlumbergerSema Spain
Albarracín 25
28037 Madrid, Spain
+34 91 440 88 00
antonio.valderrabanos@sema.es

Alexander Belskis
SchlumbergerSema Spain
Albarracín 25
28037 Madrid, Spain
+34 91 440 88 00
alexander.belskis@sema.es

Luis Iraola
SchlumbergerSema Spain
Albarracín 25
28037 Madrid, Spain
+34 91 440 88 00
luis.iraola@sema.es

ABSTRACT

This demonstration presents a tool (TExtractor) employed for enriching terminology sets in four languages: English, French, German and Spanish. We present the associated linguistic resources and the experimental results obtained in the medical domain. TExtractor has been developed within project LIQUID (IST-2000-25324), which aims at developing a cost-effective solution for the problem of cross-language information retrieval (CLIR) in multilingual document bases in technical and scientific domains.

Keywords

Natural language processing; content-based indexing and retrieval; cross-language information retrieval; terminology extraction; medical texts; gastroenterology.

1. INTRODUCTION

LIQUID aims at providing solutions to CLIR from unstructured, multilingual document bases that belong to highly specialized domains within the medical field (e.g. gastroenterology). LIQUID focuses on specialized terminology as indexing terms for two main reasons:

- terms bear most of the semantic content
- compared to general language vocabulary, terms tend to be monosemic

According to [1] terms represent best quality descriptors for document indexing due to their high informational content.

In this context, the following requirements were defined:

- affordable and feasible, i.e. the development process should be as streamlined as possible
- domain independent, i.e. portable to other domains (initially medical domains)
- language neutral, i.e. portable to other (initially Western European) languages with a reasonable effort
- complementary with existing IR systems, so it can be seamlessly integrated with them.

Proceedings of HLT 2002, Second International Conference on Human Language Technology Research, M. Marcus, ed., Morgan Kaufmann, San Francisco, 2002.

2. THE LIQUID APPROACH TO CLIR

LIQUID uses a query translation strategy in order to ensure affordability and feasibility. Using a document-translation approach requires either human or machine translation, which is expensive (in time or money) for large document collections. The main components of LIQUID are:

a) *The document base:* it contains the documents that will be accessed through the CLIR system.

b) *The term sets:* they provide the link between the documents and the queries, while connected to the concepts present in the domain ontology.

c) *The domain ontology:* it serves to structure the terminology according to its meaning and to represent the domain knowledge.

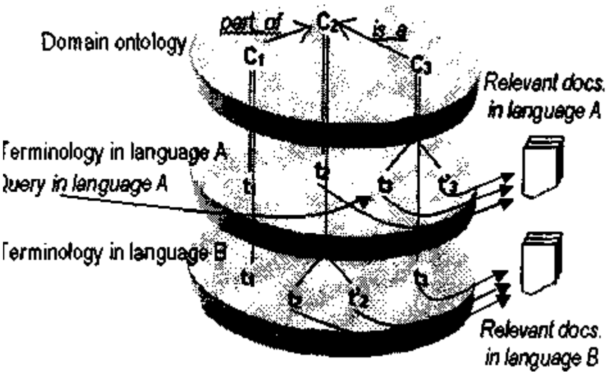

part of
C2
is a
Domain ontology
C1
C3
Relevant docs. in language A
Terminology in language A
Query in language A
t1
t2
t3
t'3
Terminology in language B
t1
t2
t'2
t3
Relevant docs. in language B

Figure 1. CLIR using a domain ontology

As a result of the combination of these three components we can link every document in the document base to the domain ontology through the set of indexing terms it contains, thus obtaining a semantic organisation of the documents. The linkage will be based on the presence of particular terms in both the semantic network and in the document. Since the semantic network is mapped to terminological resources in several languages, it is possible to make the document database available across languages.

3. TERM EXTRACTION

According to the resources used, CLIR systems can be classified in two main groups [2, 3, 4]:

- knowledge based approaches, that use multilingual glossaries and dictionaries;

- corpus based approaches, that use parallel or comparable multilingual corpora.

Examples of the first are [5] or [6]; and of the second [7]. Currently there is a tendency to combine the two approaches. The major problem for knowledge based approaches is that technical terminology is not normally present in reference works and it grows at a fast pace. Reference works hardly keep up with these new terms and then lack the necessary exhaustiveness. For corpus based approaches the problem is exactly the opposite: lack of broadness or generality. Since they are based in a particular set of texts, they are very sensitive to domain changes.

Statistical approaches can cope with high frequency terms but tend to miss low frequency terms [8], generating what's called "silence". Conversely, linguistic approaches are more efficient at identifying infrequent terms [9]. However, strategies based on linguistic knowledge tend to produce "noise" and identify as terms word combinations that are not. TExtractor uses linguistic knowledge, in the form of rules, to identify potential terms, and statistical knowledge to validate them.

Furthermore, CLIR systems need to cope with the term variation problem, i.e. the same concept can be expressed in different ways (e.g. "vaccine against HIV" and "HIV vaccine").

Our term extraction strategy is based on statistical evidence and driven by linguistic data. Linguistic analysis is based on identifying phrase delimiters and on very shallow parsing. As already stated, expensive resources like general dictionaries or full-fledged parsers, as used by [10] or [11], will not be part of our strategy in order to ensure feasibility and portability.

We will not start building new term sets from scratch, but from previously existing resources. Reusing previous efforts and ensuring coverage of most common terms are two reasons for doing so. Other researchers, like [12, 13], stress the fact that starting with a reference set improves the results of automatic term detection strategies. Since we start with a set of initial terms, the study of term variation becomes a key component [14].

3.1 Variant and New Terms

Term variation negatively affects the performance of information management systems that are unable to identify as synonyms terms that differ in their morpho-syntactic realisation (e.g. "polio vaccine" and "vaccine against polio"). The ability to detect variant and new terms will have two main benefits:

- To increase the quality of the initial term sets (which is particularly necessary when these sets do not have a wide coverage of the domain), and
- To facilitate the task of keeping the whole system (text databases and semantic networks) synchronised and updated as new documents are added.

Variant terms are terms that express the same concept as the term they derive from. They include the following types of changes or variations:

1. Morphological variations, identifying the root and its forms, like in: "X-ray therapy" and "X-ray therapies"
2. Syntactic variations in the construction of terms, like in: "HIV vaccine" and "vaccine against HIV"
3. Formal variations, like abbreviations or acronyms, like in: "PAHO" and "Pan-American Health Organization"

New terms are terms that express a different concept than the one expressed by the term they derive from. Different strategies and linguistic knowledge are employed:

1. Using known terms as a source, like extracting "common bile duct obstruction", based on "common bile duct"
2. Using suffixes, like "-itis": "diverticulitis"
3. Analysing other linguistic phenomena like co-ordination, as in the derivation of: "stomach ulcer" and "duodenal ulcer" from "stomach and duodenal ulcer"

Variant and new term generation patterns have been expressed in derivation rules. These, together with bits of linguistic knowledge, are applied by TExtractor over an initial set of seed terms in order to produce the variant and new terms.

4. TEXTRACTOR

TExtractor is a Java-based console application that allows users to produce automatically a set of new terms that are valid indexing items for a given domain. In order to achieve this goal, the tool must be provided with:

1. *Linguistic knowledge:* elementary morphological rules (stemming) plus lists of function words for each of the languages tested.
2. *Initial term sets* containing known terms for the domain and language of choice[1].
3. *Derivation rules.* Applied over known terms, they produce candidate new terms. Because of the approach followed, derivation rules are highly re-usable among languages. French and Spanish rules are almost identical, the same happens with the English and German rule sets.
4. *Validating document base* containing documents for the domain and language of choice.

All these resources are provided to TExtractor as plain text files. The results are generated also as plain text files but in order to ease their inspection, they are imported into a database (MS-Access) in which pre-defined forms and queries help users to inspect the generated terms and trace how input data has been used to produce them.

4.1 Generation and Validation

TExtractor works in two phases: generation and validation. As shown in Figure 2, the generation phase depends on two resources: an initial term set and a set of derivation rules that, applied over them, generates new terms. The initial term set is built by reusing existing glossaries; they are used as seeds or examples in order to enrich the initial term set.

The heuristic nature of the morphological derivations, the limited scope of the syntactic information (reduced to the knowledge of function words) and the absence of any semantic or contextual information, provokes over-generation. This is by no means an unexpected result, and in fact the whole approach can be viewed as an instance of the generate-and-test paradigm. Although produced according to linguistically motivated rules, many of the

[1] In the medical domain, MeSH (Medical Subject Index), SNOMED (Systematized Nomenclature of Medicine) and ICD (International Classification of Diseases) are very valuable terminological resources.

newly generated terms are not good indexing terms and should be discarded afterwards during the validation process.

Every generated term is validated against a document base that contains a substantial amount of domain-related documents. As a first validation criterion, a term is considered valid if it occurs in at least one of the documents of the base. This initial criterion can be modulated afterwards considering the frequency of appearance of the new term in the collection and/or usability constraints. For our current purposes, the criterion provides us with a reliable indication of the potential usefulness of the new term.

In order to implement the validation process, we have employed Lucene, an open source tool that provides extensive search and indexing capabilities over text files. Lucene [15] offers a well-documented interface for accessing its capabilities and we have coupled TExtractor to Lucene for checking the presence of candidate terms in our document base. Figure 3 shows the dependencies between these components:

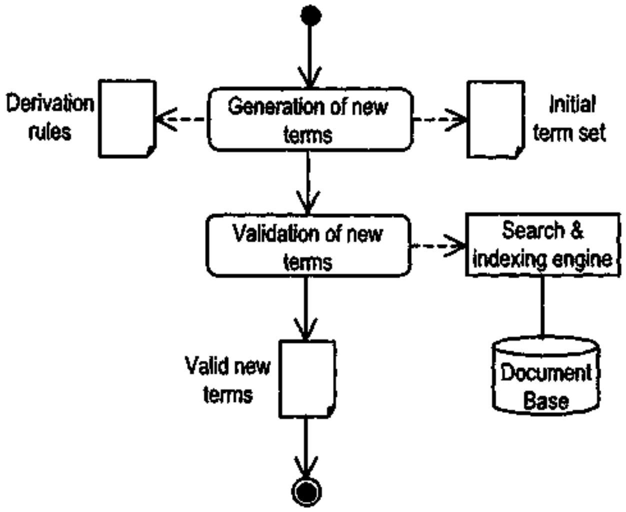


Figure 2. UML activity diagram of the term extraction and validation process

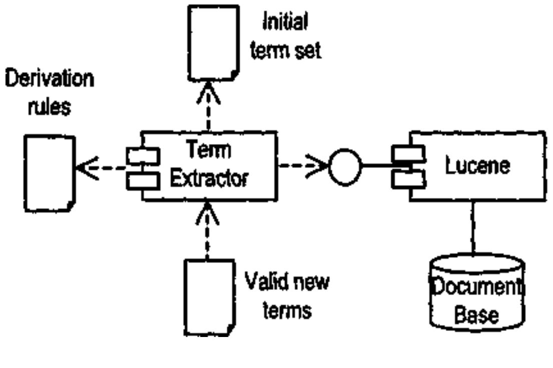


Figure 3. UML diagram of the components involved in the extraction and validation process

4.2 Derivation Rules

Rules for deriving new indexing terms conform to the classical conditional structure:

IF Antecedent Conditions THEN Consequent Actions

Antecedent conditions are checked on a singular term (a member of the set of initial terms) and, if fulfilled, applying the sequence of consequent actions over it produces a newly generated term. Both conditions and actions apply over the individual tokens that compose a typical multi-word term.

4.2.1 Rule syntax

Derivation rules consist of three clearly differentiated parts:

1. *Rule identifier.* Typically a brief phrase describing the derivation pattern implemented in the rule. It is used for tracking which rule has been applied for deriving a particular term. The rule identifier is separated from the rule proper by a colon sign.
2. *Antecedent.* A sequence of blank-separated token identifiers. Each token identifier is represented by the characters "tr" plus an ordering number that identifies each of the tokens that appear in a single rule. Bounded and unbounded repetition operators (+, *, ?) can be added to token identifiers, thus allowing the codification of highly flexible input term patterns. The antecedent ends with the greater than (">") symbol, that separates it from the consequent.
3. *Consequent.* A sequence of blank-separated token identifiers (a subset of the tokens mentioned in the antecedent) and/or actions over those tokens.

4.2.2 Antecedent conditions

The kind of conditions that can be imposed over any individual token fall under one of the following categories:

1. Typographical, such as the presence of a hyphen or an initial capital in the token.
2. Morphological, such as the property of number for nouns.
3. Syntactic, such as the part-of-speech of function words.

Morphological properties are determined by means of simple suffix checking and applying highly productive heuristics. Because of their simplicity and quick development, these mechanisms are cost-effective and scalable to most European languages. Although cases of over-generation do occur, they are discarded in the subsequent validation phase. This approach could also be described as a form of stemming [16].

As an example of the kind of morphological processing done in TExtractor, what follows is an excerpt of the singular-plural Spanish morphology:

```
#Singular; Plural for Spanish
DEFINITION_BEGIN
#casa ; casas
a ; as
#tahalí ; tahalíes
í ; íes
#verdad ; verdades
d ; des
#vez ; veces
z ; ces
#análisis ; análisis
sis ; sis
```

The second example is taken from the set of English suffixes employed for deriving medical terminology:

```
#Suffixed; Resuffixed for English
# anaesthesia ; anaesthetic
esia ; etic
etic ;  esia
# cardiography ; cardiogram
graphy ; gram
gram ; graphy
# microscope ; microscopy
scope ; scopy
scopy ; scope
...
```

Regarding part-of-speech determination, and following the general approach towards cost-effectiveness and scalability, only function words (closed categories) are considered. Conditions imposing the membership to an open POS category are automatically granted, as in the following rule that derives a new term if the initial one is a singular noun:

```
plural: tr1[Noun, Singular] > MakePlural(tr1)
```
lobotomy > lobotomies

Given that TExtractor does not have in its current state any means for tagging "lobotomy" as a noun, the rule fires anyway and the plural form is derived. POS tagging of open categories has been included in the rules mainly to improve their readability, since no wide coverage mechanism for POS determination has been incorporated. On the other hand, words belonging to closed categories have been compiled in lists and are available for checking tokens in rules:

```
conjunction_right_part:
tr1+ tr2[Class:ConjunctionCopulative] tr3+ > tr3
```
Head and neck neoplasms > neck neoplasms

4.2.3 *Consequent actions*

Consequent actions apply over individual tokens identified in the antecedent, as in the previous example where the action "MakePlural" is applied over token number one.

Consequent actions may affect to several tokens of the term, such as when:

1. Re-ordering the token sequence.
2. Joining two tokens in a single one.

Or may affect to an individual token, such as when:

3. Removing/inserting a certain token.
4. Modifying the typographical, morphological and/or syntactic properties of a certain token.

4.3 Experimental Results

As an initial test of TExtractor, we have taken the ELCANO document base, a publicly available collection of clinical cases in the domain of gastroenterology (http://www.imim.es/elcano). This base includes medical articles belonging to that domain and written in English and Spanish[2]. Each article is provided with several indexing terms (keywords) also in English and Spanish. We have written 67 derivation rules for English and the same number of rules for Spanish. Most of the rules are identical for both languages and the differences when exist are mostly due to syntactic differences between both languages in the structure of noun phrases.

Table 1. Initial resources employed

Language	# documents	# initial terms	# rules
English	563	1222	67
Spanish	563	1226	67

Newly generated terms have been automatically validated against the ELCANO document base, thus considering valid new terms to those new terms that occur in at least one document. The same criterion has been applied for validating the initial term sets, so producing a subset of valid initial terms (i.e. those keywords that do occur in at least one document). This initial criterion can be modulated afterwards considering the frequency of appearance of the new term in the collection and/or usability constraints. For our current purposes, the criterion provides us with a reliable indication of the potential usefulness of the new term.

Our main performance metric is the ratio between valid new terms and valid initial terms. It gives us a quantitative measure of how successfully we have enriched the initial term set.

Table 2. Performance metric

Language	# valid initial terms	# valid new terms	enrichment ratio
English	874	525	60%
Spanish	888	800	90%

In order to verify the quality of the automatic validation process, the set of valid new terms has been manually checked. New terms in both languages have been reviewed, looking for *spurious terms* (syntactically ill-formed terms, such as *"infection in surgical")* and *irrelevant terms* (generic non-medical terms such as *"History" or "expert").*

Table 3. Quality inspection results

	English	Spanish
Number of spurious terms	13 (3%)	8 (0,94%)
Number of irrelevant terms	8 (1,14%)	27 (3,19%)
Spurious plus irrelevant terms	21 (4,14%)	35 (4,13%)

4.4 TExtractor Graphical Interface

The graphical interface of TExtractor responds to the necessity of easily demonstrating its capabilities and also to allow users unfamiliar with the intricacies of the application to run it and inspect its results.

[2] Work on French and German is ongoing: while results are still preliminary, enrichment percentages are encouraging. We are in the process of incorporating the most prestigious multilingual terminological resource for our domain: *The International Wordbook of Gastroenterology,* Pounder, R. & M. Hudson, Radcliffe Medical Press, 1994.

Given that TExtractor is a Java-based console application that takes as input many text files and produces as output several others, its use requires a certain degree of familiarity with its operation parameters and specially with the location and name of all its input and output files. The graphical interface attempts to hide all those details from its demonstrator or its occasional user, allowing them to focus on the design and performance of the tool instead of on the minute details of its operation.

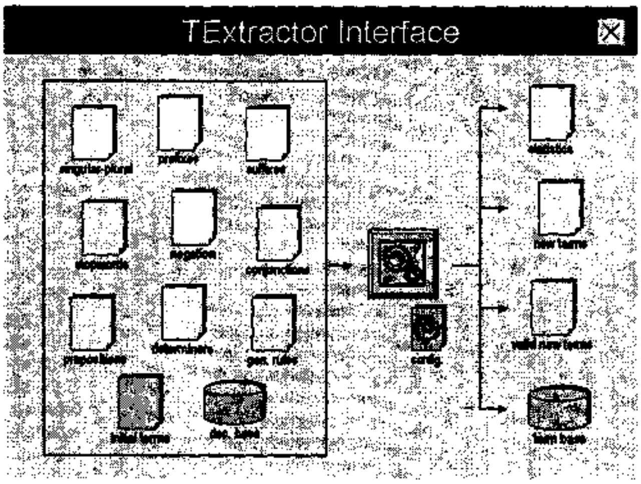
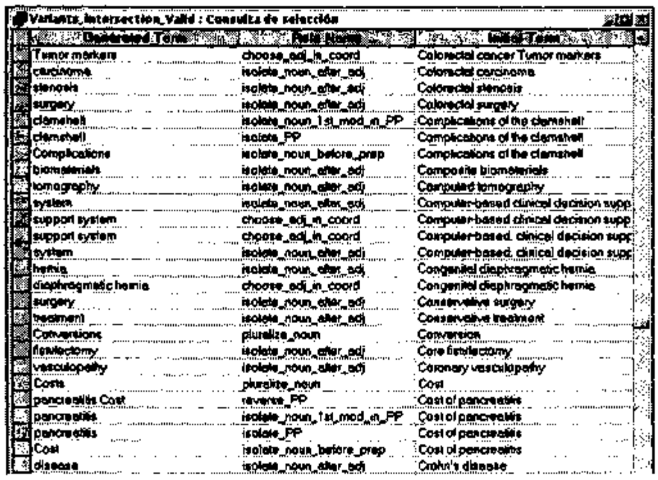


Figure 4. Screenshot of the TExtractor graphical interface

Input and output text files are represented as document icons. Each icon can be activated by clicking onto it. When activated, a text editor is invoked and the represented text file loaded in it. The user may then view, edit and save the file contents.

The TExtractor application is represented as a central button. Pressing it opens a system console were the Java application runs. When the application ends the console is also closed and the user may then inspect the output files. The document and term bases can also be accessed in the same way as text files, except that instead of a text editor, MS Access is used to view them.



Variants_Intersection_Valid : Consulta de selección

Generated Term	Rule Name	Initial Term
Tumor markers	choose_adj_in_coord	Colorectal cancer Tumor markers
carcinoma	isolate_noun_after_adj	Colorectal carcinoma
stenosis	isolate_noun_after_adj	Colorectal stenosis
surgery	isolate_noun_after_adj	Colorectal surgery
clamshell	isolate_noun_1st_mod_in_PP	Complications of the clamshell
clamshell	isolate_PP	Complications of the clamshell
Complications	isolate_noun_before_prep	Complications of the clamshell
biomaterials	isolate_noun_after_adj	Composite biomaterials
tomography	isolate_noun_after_adj	Computed tomography
system	isolate_noun_after_adj	Computer-based clinical decision supp
support system	choose_adj_in_coord	Computer-based clinical decision supp
support system	choose_adj_in_coord	Computer-based clinical decision supp
system	isolate_noun_after_adj	Computer-based clinical decision supp
hernia	isolate_noun_after_adj	Congenital diaphragmatic hernia
diaphragmatic hernia	choose_adj_in_coord	Congenital diaphragmatic hernia
surgery	isolate_noun_after_adj	Conservative surgery
treatment	isolate_noun_after_adj	Conservative treatment
Conversions	pluralize_noun	Conversion
fistulectomy	isolate_noun_after_adj	Core fistulectomy
vasculopathy	isolate_noun_after_adj	Coronary vasculopathy
Costs	pluralize_noun	Cost
pancreatitis Cost	reverse_PP	Cost of pancreatitis
pancreatitis	isolate_noun_1st_mod_in_PP	Cost of pancreatitis
pancreatitis	isolate_PP	Cost of pancreatitis
Cost	isolate_noun_before_prep	Cost of pancreatitis
disease	isolate_noun_after_adj	Crohn's disease

Figure 5. Excerpt from a list of new valid terms along with the rule names and initial terms that generated them.

5. CONCLUSIONS AND FURTHER WORK

We have presented TExtractor, a solution to the problem of cross-lingual access to multilingual document bases in specific domains. The solution involves a language independent domain ontology and a terminology extraction tool that provides to the ontology linguistic realisations of domain concepts in four languages: English, French, German and Spanish.

In this paper we have focussed on the terminology extraction tool, Textractor. We have shown that it is possible to enrich substantially an initial set of indexing terms applying a generate-and-test framework. Our approach to term extraction can be characterised by:

- Very low dependency on linguistic resources.
- Small set of linguistically motivated derivation rules.
- Incorporation of publicly available software tools.
- Exhaustive validation of the newly generated terms against a domain document base.

This approach has been tested in the domain of gastroenterology with a collection of documents and an initial set of indexing terms, both in English and Spanish. In further work, we will pay attention to issues such as:

- Re-use of derivation rules. Attempting to capture language independent derivation phenomena.
- Incorporation of publicly available, wide coverage linguistic resources that will enhance the derivation capabilities while maintaining the overall cost-effectiveness and scalability.
- Incorporation of publicly available terminological resources in the medical domain and for the languages considered in the project.

6. REFERENCES

[1] Lewis, D. and Croft, W. "Term clustering of syntactic phrases". In ACM SIGIR-90, 385-404. 1990.

[2] Gonzalo, J., F. Verdejo and I. Chugur. "Using EuroWordNet in a Concept-Based Approach to Cross-Language Text Retrieval". Applied Artificial Intelligence Special Issue on Multilinguality in the Software Industry: the AI contribution. 1999.

[3] L. Ballesteros and W.B. Croft. "Phrasal Translation and Query Expansion Techniques for Cross-Language Information Retrieval", in Proceedings of ACM SIGIR Conference, 20: 84-91. 1997.

[4] Jacquemin, C., Bourigault, D. "Term Extraction and Automatic Indexing". In R. Mitkov, editor, Handbook of Computational Linguistics. Oxford University Press, Oxford. 2000.

[5] Hull, D., and Grefenstette G. "Experiments in Multilingual Information Retrieval". Proceedings of ACM, SIGIR'96. Zurich. 1996.

[6] L. Ballesteros and W.B. Croft. "Phrasal Translation and Query Expansion Techniques for Cross-Language

Information Retrieval", in Proceedings of ACM SIGIR Conference, 20: 84-91. 1997.

[7] Sheridan, P. and Ballerini, J. P.. "Experiments in multilingual information retrieval using the spider system". In Proceedings of ACM/SIGIR. 1996.

[8] Evans, D., and Chengxiang Zhai. "Noun-Phrase Analysis in Unrestricted Text for Information Retrieval". Proceedings, 34th Annual Meeting of the Association for Computational Linguistics, 17-24. 1996.

[9] Bourigault D. "An endogenous corpus-based method for structural noun phrase disambiguation". In Proceedings of the 6th Conference of the European Chapter of the Association for Computational Linguistics (EACL'93). Utrecht, The Netherlands. 1993.

[10] Arppe, Antti. "Term Extraction from Unrestricted Text". Paper presented at NODALIDA-95, Helsinki (Available at http://www.lingsoft.fi/doc/nptool/term-extraction-html - 20-12-00). 1995.

[11] Justeson, J. and Kate, S. (1995) "Technical Terminology: Some Linguistic Properties and an Algorithm for Identification in Text", in Natural Language Engineering, Vol. 1, No 1: 9-27

[12] Jacquemin, C., Klavans, J., Tzoukermann, E. "Expansion of multi-word terms for indexing and retrieval using morphology and syntax". Proceedings, 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics (ACL-EACL 97), 24-31. Madrid. 1997.

[13] Jacquemin, C., and Tzoukermann, E. "NLP for Term Variation Extraction: a Synergy of Morphology, Lexicon and Syntax". In T. Strzalkowsky, editor, Natural Language Information Retrieval, 25-74. Kluwer. Boston, MA. 1999.

[14] Daille, B., Habert, B., Jacquemin, C., and Royauté, J. Empirical observation of term variations and principles for their description. Terminology, 3(2), 197-258. 2000.

[15] Goetz, Brian. "The Lucene search engine", Java World. The official Lucene web site is located at: http://jakarta.apache.org/lucene. 2000.

[16] M.F.Porter. "An algorithm for suffix stripping", Program, 14(3):130-137. 1980.