

# LINGUISTIC ENGINEERING APPROACH TO THE ENHANCEMENT OF WEB-SEARCHING

Ana García-Serrano<sup>1</sup>, Paloma Martínez<sup>2</sup> and Alberto Ruiz<sup>1</sup>

(1) Department of Artificial Intelligence, Technical University of Madrid  
Campus de Montegancedo 28660 Boadilla del Monte, Madrid, Spain  
email: {agarcia, aruiz}@dia.fi.upm.es

(2) Computer Science Department, Carlos III University  
Avda. Universidad 30, 28911 Leganés, Madrid, Spain  
email: pmf@inf.uc3m.es

## Abstract

This paper describes a proposal to improve the web document retrieval by facing the main three problems in document searching: first, traditional web search engines miss documents that are relevant to the user query and retrieve many that are not. Second, the query formulation is not as accessible as it could be, and some users have difficulties in expressing boolean queries. Third, the result sorting is usually arbitrary or based on statistical data. The on-going research presented in this paper focuses on the use of a natural language interface for interacting with search engines, both before and after the search process. This interface performs enrichment of user queries based on lexical resources, and makes use of semantic domain information for expanding and sorting the results. Preliminary experimental work and discussion of the obtained results are also outlined.

## Keywords

lexical resources, document searching, natural language technology, domain ontology.

## 1 Introduction

The high amount of unstructured information underlying the web, as well as the complexity of web browsing and searching applications are demanding the use of linguistic technologies to the development of real domain systems in the context of information technology.

The web site of an official institution has as main objective to inform the citizens; thus, to provide an accurate search engine as well as an adequate presentation of retrieved results (according to variable criteria) are two crucial aspects for the user approval in any organization that uses an Internet based information system.

There are several problems with the current search engines: many of the documents retrieved for general queries are irrelevant to the subject of interest and other documents are missing because the query does not include the exact keywords; users are not confident about the languages used to formulate their queries; refined queries with restrictive boolean operators may result in a few or even no documents, etc. This motivates the use of natural language interfaces as an adequate way of communicating with search engines.

To improve the quality of the search on Internet, two main approaches have typically been adopted:

1. Creation of a metasearch engine that makes use of multiple search engines (this implies the unification of both the query language and the type of results returned by the different search engines).

2. To use NLP techniques (query extensions and improving the quality of information retrieved using NLP-based systems) due to traditional keyword-based retrieval presents several inadequacies, [1], such as the impossibility of handling morphological, lexical, semantic and syntactic variations.

Focusing on the second issue, we present the on-going research project, called MESIA<sup>1</sup>, for the Madrid Local Government web site ([www.comadrid.es](http://www.comadrid.es)) whose main goal is to enhance the answers provided by AltaVista search engine through the extension of user queries and the filtering of answers. This web site includes information about several topics: education, sports, culture and leisure and so on.

Our proposal is to provide linguistic mechanisms that transform and extend the user questions by integrating available general-purpose Spanish lexical resources: ARIES morphological database, [2], shallow parsing, [3], and EuroWordNet semantic

<sup>1</sup> Spanish acronym for "A computational model for sensitive information extraction from short texts". This work was supported by MESIA project CAM 07T/0017/1998.

database, [5]. Additionally, an ontology for storing document metadata which define a set of terms and relationships that characterize a domain area is being used.

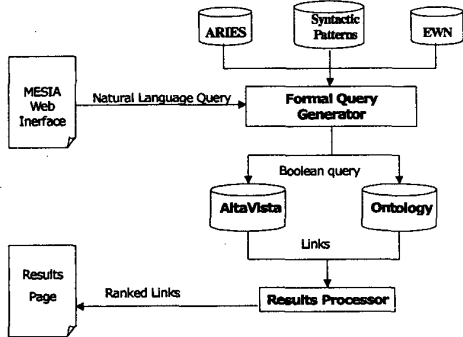


Fig. 1: MESIA architecture

A layered NLP component has been installed on a proprietary search engine (Altavista, which is used as is). The current prototype is running in Ciao Prolog, [6], and Java.

Next section presents the approach followed to handle natural language user queries as well as a brief description of the lexical resources used. Section 3 is devoted to describe the ontology used in a specific knowledge domain that contributes to rank the documents retrieved. Finally, an evaluation of the experiments carried out is presented as well as some conclusions and future work.

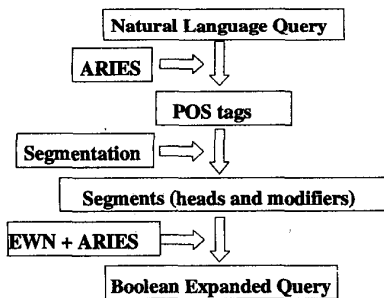


Fig. 2: Layered architecture of the Formal Query Generator

## 2 Extending web queries using lexical resources

Figure 1 shows an overview of MESIA architecture. Focusing on the Formal Query Generator component, it deals with morphological and lexical variation of significant words of user queries in order to enhance document searching. It is well known that morphological and lexical variation hurts recall in information retrieval (IR), [1].

In order to handle morphological variation we have integrated ARIES ([www.mat.upm.es/~aries/](http://www.mat.upm.es/~aries/)), [2], a Spanish lexical platform developed by the Universidad Politécnica de Madrid and Universidad Autónoma de Madrid. ARIES is composed of a Spanish lexicon with around 38,000 lemma entries, including 21,000 nouns, 7,300 verbs, 10,000 adjectives and around 500 entries for prepositions, conjunctions, articles, adverbs and pronouns; some access utilities and a morphological analyzer/generator are also included. The morphological analyzer assigns part-of-speech tags to the query words (useful to identify the relevant terms of the query). Moreover, a DCG morphological generator for deriving word variants is being incorporated in MESIA system. This generator allows, for instance, obtaining number and gender forms from a nominal lemma. An example of an ARIES lexical entry (*doctor*) is shown below:

```
doctor
category=n /* noun */
concat=wl /* it accepts a number morpheme */
agr gender=masc /* masculine gender */
agr number=sing /* singular number */
plural derivation=plu2 /*rule for plural generation */
lex =doctor /* lemma */
```

Lexical variation is accomplished by using EuroWordNet (EWN) [www.hum.uva.nl/~ewn/](http://www.hum.uva.nl/~ewn/), [5], a lexical database that is structured as a top concept ontology that reflects different explicit relationships. Words are organized in synonym sets, called *synsets* (each synset represents a concept). The synsets are related by hyponymy and hyperonymy (IS-A) relationships. Spanish database approximately contains 24.000 nouns and 4.100 verbs.

EWN permits dealing with lexical (discriminating word senses in documents and queries) and semantic variations (matching semantically related words). If terms are fully disambiguated it should improve precision, and if equivalent terms are identified it would improve recall. It seems that dealing with lexical variation is more beneficial for incomplete and relatively short queries, [7].

Figure 2 shows the layered architecture of the Formal Query Generator module. In a first step the natural language query is tokenized, and ARIES morphological analyzer assigns the possible part-of-speech (POS) tags to each query word. Then, a segmentation process is performed in order to solve the ambiguity produced by ARIES part-of-speech analyzer. It also detects the query segments; this task is carried out by a simple phrase segmenter based on cascade finite automata, [3], which identifies simple noun, prepositional and verb phrases in the query. This shallow parsing also extracts the significant query terms or *keywords* (phrase heads and modifiers)

that have to be extended by using ARIES and EWN lexical resources.

With the lemma extracted for each keyword and taking into account its grammatical category, EWN is used for extracting semantically related terms; in a first approach only synonym words are used. For instance, given the noun “estancia” (*stay*), EuroWordNet provides:

```
?-ewn(estancia,n,Synonyms,_,_).
Synonyms= syn([permanencia]),
Synonyms=syn([cortijo, labranza,
heredad, hacienda, granja])
```

Then, ARIES morphological generator provides the morphological variants for original query keywords and also for those proceeding from EWN. Nominal and adjectival keywords are then expanded with gender and number variations and verbal keywords with their corresponding infinitive lemma. This step is necessary, as the search engine does not perform any kind of stemming/morphological inflection for Spanish language. For example, given the noun “*doctor*” ARIES returns four morphological variants: *doctoras, doctores, doctora, doctor*.

Keywords and their variants are converted into a conjunct of disjuncts (conjunctive normal form) [17,9,4].

In short, the strategy for boolean query generation is this: semantically related terms are added to each relevant term connected by ORs. For instance, the term “*becas*” (grants) is expanded as: (*beca OR galardón OR apoyo OR ayuda*).

Then, morphological variations are added to each term connected by ORs. For instance, the previous terms are expanded as: (*becas OR beca OR galardones OR galardón OR apoyos OR apoyo OR ayuda OR ayudas*). Finally, all the sets of expanded terms are connected by the AND operator. The user query *¿qué becas postdoctorales hay?* (*Which postdoctoral grants are there?*) is translated into: (*becas OR beca OR galardones OR galardón OR apoyos OR apoyo OR ayuda OR ayudas*) AND (*postdoctorales OR postdoctoral*).

This query is ready to be used as input both for the web search engine and the domain ontology access, as will be explained in the next chapter. Figure 3 shows the MESIA interface, on which the different query extensions are displayed<sup>2</sup>.

As it has been described, the EWN query enlargement adds all the synonyms of a keyword in a first approach. The retrieval process itself performs a disambiguation (the conjunction of terms, as a restriction, eliminates many of the spurious forms). Next subsection outlines several proposals

concerning query extensions that make use of different knowledge sources, as well as different approaches to cope with the ambiguity problem.

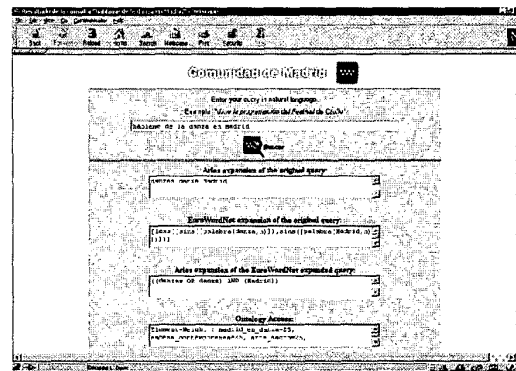


Fig. 3: MESIA interface at the Madrid Local Government web site

## 2.1 Related work

In [7] is investigated the effect of expansion by selecting query concepts to be expanded by hand with synonyms extracted from WordNet, [8]; works presented in [9] and [4] are proposals for query enrichment use morphology and a hierarchical thesaurus to generate boolean queries. In [10] is proposed a layered approach to convert English queries for use by the various search engines: morphological analysis, noun phrase syntax and semantic expansion based on EWN. An approach for query expansions by refining the set of documents used in feedback is described in [11].

Concerning the word sense disambiguation (WSD) problem that appears when semantically related terms are added to the original query terms, it affects precision values. In [13] are described some works that use Wordnet IS-A and other semantic relations for WSD. A recent work is [12] that propose WSD using the IS-A relations of WordNet by means of synsets that subsumes sentence word senses and other heuristics such as the use of Wordnet definitions; a method for WSD in query extensions in two steps: ranking from Internet and semantic density with WordNet hierarchy and glosses is defined in [17]. However, WSD has usually poor results; reasons for this lack of success are described in [14] explaining that mistakes in disambiguation could be worse than the original ambiguity (WSD was only of use in a retrieval context if queries were short or if disambiguation was performed at a high level of accuracy). Query word collocation has a strong effect and senses of many words have a skewed frequency distribution as is exposed in [16]. Finally, indexing

<sup>2</sup> [http://tornado.dia.fi.upm.es/mesia/mesia\\_demo.html](http://tornado.dia.fi.upm.es/mesia/mesia_demo.html)

by word senses prevents some matching that can be useful for retrieval (grouping of senses that do not typically extend across grammatical categories), [15].

### 3 A domain ontology model to sort and expand the results

We may suppose that if the user is querying for one issue, s(he) will also be interested in related subjects. These related issues are not necessarily described by the same keywords as the original one: in this case, the keyword-based search cannot improve the accuracy of the retrieval, even after the linguistic expansion.

Besides, when it comes to sorting the result links, there are no adequate criteria to do it properly, other than statistical data or past queries historical archives.

With the use of domain information, the query subject can be deduced and then the results can be expanded to similar issues, not depending on keywords but on the conceptual domain knowledge stored in the ontology.

This enrichment could be designed as another step of the query expansion, but the MESIA approach is different: the ontology access and the web search are launched at the same time and they work independently, both using the expanded query as input. The advantage of having these two resources working independently is that if one of them is not available at the query time for any reason, then the other will remain operative and the user will get some results.

When both processes are finished, the Results Processor gets the results and merges them. The results sorting is made according to the closeness of every link to the user query; this information is retrieved from the ontology.

As the MESIA domain is not very large, its ontology was designed as a tree structure. The hierarchical structure simplifies the ontology access; for larger domains, a more complex structure like a semantic net could be needed.

The ontology has proved to be the best way to represent a set of concepts and the relationships between them [19].

Figure 4 represents the part of the domain ontology that is already working in MESIA

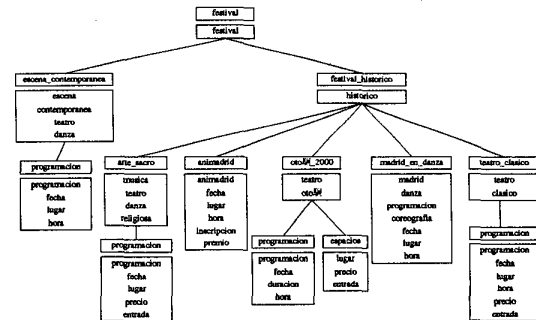


Fig.4: Domain ontology

The MESIA ontology was implemented in XML. This markup language provides a simple way to describe structured knowledge; besides, it is text based, and therefore easy to be modified. Although XML does not facilitate data processing (such as inheritance between ontology nodes), it is one of the most used languages for ontology support.

Every node in the ontology represents an issue, storing the following data:

1. A descriptive name of the issue, to be displayed before the related links.
2. A set of keywords regarding the issue.
3. A list of related links.

For example, the "theater\_festival" node would have "Theater Festivals in Madrid" as a title, and its keyword list would include terms like "theater", "festival", "representation", "show" or "play". As lower nodes in the structure tree inherit these keywords, a term should not be associated to a node if it does not apply to all the nodes below.

The desired role for the ontology in the search process determines the use of the related links field in the ontology nodes; two approaches can be considered:

1. The ontology access retrieves and sorts links: Every node stores all the links related to the issue. In order to achieve this, a previous classification of all the web documents in the domain is needed. Then, the web search is completely replaced by an ontology access: the links will be retrieved from the nodes whose issue is closer to the query subject. Once the related nodes are identified, other nodes that are near in the structure tree are also considered, leading to a non-keyword based semantic enrichment of the results. As all the links stored in a node are relevant to the node issue, the success of the search process will only depend on the accuracy and significance of the node keywords. Unfortunately, the ontology maintenance requires a significant effort, as every change in a link involves an ontology update. If the domain is large, this choice is not viable.
2. The ontology only sorts the links retrieved by the web search: The nodes do not store links; once the

documents are retrieved from the web, they are automatically classified *on the fly* by accessing the ontology. Then, they are sorted and presented to the user. The sorting method will be explained later. The ontology maintenance cost is low, as it is only needed to keep the ontology updated with the issues that can be found in the domain, providing the proper keywords for them. However, as the nodes do not store links, there is no result expansion. It could only be achieved by expanding the query with node keywords and then performing the web search.

Both approaches have to deal with the automatic or semiautomatic classification of links; however, as the domain considered in MESIA is relatively small, the current prototype works following the first approach, storing in the ontology every relevant link. The problem is avoided by performing a previous, manual classification of the domain web pages. In a further development step, the automatic classification problem will be faced.

As the classification of a link in the ontology is based on a set of keywords describing its content, the problem can be reduced to obtain a set of significant words from a given web document. This constitutes a text mining problem [18]. As this falls out of the bounds of the MESIA project, and given the moderated size of the domain, the web developers in this project are asked to include a metadata HTML field in their documents. This special field just includes a few words describing the document content that will be used for the classification of the document in the ontology. For example:

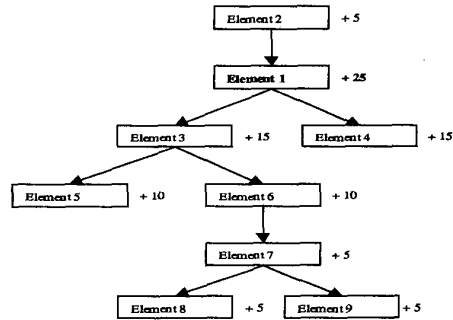
```
<meta name="mesia" content="theater november
play Madrid">
```

The ontology access is based on a weight system. The weight is a numerical value that measures the relevance of an ontology node for a given query.

The access method works as follows: every word in the expanded query is matched with all the keywords of the ontology nodes. If a query word is found in one or more nodes, this node gets its weight increased. Besides, all the nodes below it inherit that weight, but with a lower numerical value. The parent node also gains weight.

When this process is finished, those nodes that received some weight are sorted according to their value. If a node did not gain weight, it will not appear in the display of results.

Figure 5 represents the current weight values working in the MESIA system.



This system applies for the two main kinds of ontology access:

1. Given a query, the ontology access returns a sorted list of issues related to the subject, according to the weight received by each issue. If the ontology nodes store links, these links are displayed as results. If not, they will be useful to sort the links retrieved from the web.

2. Given a metadata field containing the descriptive words of a web document, it classifies the document in one of the existing nodes: that which obtained the highest weight.

The ontology node management is not the issue of this paper, but it can be easily handled by an interface, given the simplicity of the XML structure. Eventually, the inclusion of new issues in the domain may require the creation or destruction of ontology nodes.

#### 4 Some experimental results

In order to evaluate how the linguistic knowledge affects the retrieval results, a first study of seven queries in four types of experiments, [20], concludes that the use of ARIES morphology generally enhances the search results. However, the extension with synonyms/hyperonyms<sup>3</sup>, although contributes to retrieve documents that are not retrieved in the other experiments, affects the precision measure. In a second evaluation work we have run 20 short user queries (from 3 to 10 words) in the same four types of experiments, all of them executed in the Altavista Advanced Search mode. These experiments are: (1) baseline (the relevant words of the query linked by AND operator), (2) only morphological variations expansion (with ARIES), (3) only semantic expansion with synonyms (with EWN) and (4) expansion with ARIES and EWN.

Modifications performed to the first experiment were: only verbal infinitive lemma is added when a query keyword is a verb, only synonyms are included considering that they are obtained taking into account

Fig. 5: An example of weights in domain ontology

<sup>3</sup> In this experimental work, hyperonyms of query keywords were also included.

the keyword POS tag and, finally, to use *Order by field* with original relevant terms in order to obtain relevant documents ranked uppermost. Precision values are measured considering relevant documents in 20 top ranking due to users uniquely consider the uppermost retrieved documents. Average precision obtained in the four experiments was: (1) 55%, (2) 66%, (3) 68% and (4) 77%. These results show that a combination of morphologic and semantic features could enhance information retrieval. However, by examining separately some queries it can be observed that EWN expansion affects precision.

## 5 Conclusions and future work

In some cases, expansion with EWN/WordNet must be constrained or precision will suffer drastically. However, EWN/WordNet poses some problems for IR, [15, 17]; EWN/WordNet does not include cross-part-of-speech semantic relations; too much fine-grained sense distinctions (the degree of granularity required is task-dependent); lack of domain information (it would be very interesting to manage different domains that allow to store semantic preferences depending on the activated domain).

There can be established important contributions for interactive systems intended for casual users that tend to formulate short queries; an adequate proposal is to give the user the possibility of specifying the query related terms from those derived from EWN and, in this way, to prepare different query extensions to be displayed to the user.

As future enhancements we propose to use weighting information (depending on if IR system accepts it); new lexical operators such as paragraph, sentence and so that appear in [17].

## References

1. A. T. Arampatzis, Th.P. van der Weide, P. van Bommel and C. Koster, Linguistically-motivated Information Retrieval, TR CSI-R9918, Sept. 1999.
2. J. M. Goñi, J. C. González and A. Moreno, ARIES: A lexical platform for engineering Spanish processing tools. Natural Language Engineering, vol 3 no 4, pp. 317-345, 1997.
3. P. Martínez and A. García-Serrano, The role of knowledge-based technology in language applications development. Expert Systems with Applications, vol. 19, no 2, pp. 155-160, 2000.
4. A. F. Gelbukh, Lazy query enrichment: a method for indexing large specialized document bases with morphology and concept hierarchy, Proc. DEXA 2000.
5. P. Vossen, The EuroWordNet Base Concepts and Top Ontology. Version 2. EuroWordNet (LE 4003) Deliverable, 1998.
6. F. Bueno, D. Cabeza, M. Carro, M. Hermenegildo, P. López, and G. Puebla, The Ciao Prolog System: A Next Generation Logic Programming Environment, TR CLIP 3/97.1(www.clip.dia.fi.upm.es/Software/Ciao/), 1999.
7. E. Voorhees, On expanding query vectors with lexically related words, 2<sup>nd</sup> Text Retrieval Conference, pp. 223-231, 1994.
8. C. Fellbaum (ed.), WordNet: An Electronic Lexical Database, MIT Press, 1998.
9. J. Cowie, Collage: An NLP toolset to support boolean retrieval, Natural Language Information Retrieval, T. Strzalkowski (Ed.), Kluwer Academic, 1999.
10. S. Flank, A layered approach to NLP-based Information Retrieval, Proc. COLING 1998.
11. M. Mitra, A. Singhal and C. Buckley, Improving Automatic Query Expansion, Proc. 21<sup>st</sup> ACM-SIGIR Conf. on Research and Development in Information Retrieval, 1998.
12. A. Montoyo and M. Palomar, WSD algorithm applied to a NLP system, M. Bouzeghoub et al. (Eds): NLDB 2000, LNCS 1959, pp. 54-65, 2001.
13. N. Ide and J. Véronis, Word Sense Disambiguation: the State of the Art, Computational Linguistics, vol 24 no 1, 1998.
14. M. Sanderson, Retrieving with good sense, Information Retrieval Journal, vol 2 no 1, pp. 45-65, 2000.
15. J. Gonzalo, F. Verdejo, I. Chugur and J. Cigarran, Indexing with WordNet synsets can improve Text Retrieval, Proc. COLING/ACL '98 Workshop on Usage of WordNet for NLP, Montreal, 1998.
16. R. Krovetz and W.B. Croft, Lexical Ambiguity and Information Retrieval, ACM Transactions on Information Systems, vol 10 no 1, 1992.
17. R. Mihalcea, Word Sense Disambiguation and its application to Internet search, Master Thesis School of Engineering and applied Science, Southern Methodist University, 1999.
18. A. Maedche and S. Staab. Semi-Automatic Engineering of Ontologies from Text. Proc. Of the 12<sup>th</sup> Int. Conf. on SW and KW Engineering, 2000.
19. A.Gómez-Pérez, Knowledge Sharing and Reuse. J. Liebowitz (Ed.) Handbook of Expert Systems. CRC, 1998.
20. A. García-Serrano and P. Martínez. An interface Agent with linguistic skills. Proc. NLDB 01, 2001.