The query answering system PRODICOS

Laura Monceaux, Christine Jacquin, and Emmanuel Desmontils

Université de Nantes, Laboratoire LINA,
2 rue de la Houssinière, BP92208,
44322 Nantes cedex 03

Abstract. In this paper, we present the PRODICOS query answering system which was developed by the TALN team from the LINA institute. We present the various modules constituting our system and for all of them, their evaluation in order to explain the obtained results. Then, we present the main improvement based on the use of semantic data.

1 Introduction

In this paper, we present the PRODICOS query answering system which was developed by the TALN team from the LINA institute. It was our first participation to the CLEF evaluation campaign. We have decided to participate to the monolingual evaluation task dedicated to the french language. This campaign enables us to analyse the performances of our system. Firstly, we present the various modules constituting our system and for all of them, their evaluation in order to explain the obtained results. Secondly, we present the expected improvement based on used of semantic data: the EuroWordnet thesaurus [1] and topic signatures [2].

2 Overview of the system architecture

The PRODICOS query answering system is divided into three parts (figure 1):

- question analysis module;
- sentence extraction module (extracts sentences which might contain the answer);
- answer extraction module (extracts the answer according to the results provided by the previous module).

The modules of the PRODICOS system are based on the use of linguistic knowledge, in particular lexical knowledge coming from EuroWordnet thesaurus [1] and syntactic knowledge coming from a syntactic chunker which has been developed by our team (by the use of the TreeTagger tool [3]).

The system has participated to the CLEF 2005 evaluation campaign for the monolingual query answering task dedicated to the french language. This campaign enables us to make a first evaluation of the system. It allows us to compute the performances of the various modules of the system in order to analyse their

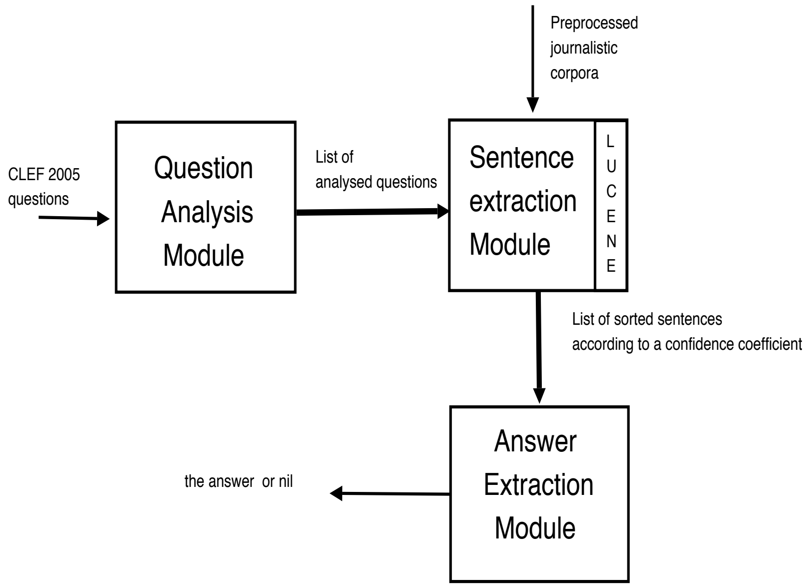

Fig. 1. The PRODICOS System

weaknesses and the possible need of semantic knowledge. We present, in the next sections, more in detail, the various modules which belong to the PRODICOS system and linguistic tools used to implement them. In parallel, we analyse in detail the results of each system module.

3 Question analysis module

Question analysis module aims at extracting features from questions that will be used in different PRODICOS's system modules, in particular to define the patterns to search the right answer. In our system, the question analysis module provides several pieces of information:

- a question type in order to associate to the question a list of patterns for extracting the answer. Indeed, searching the right answer to a definition question like "Qu'est ce que les FARC ?" will be completely different as performing the same search on a factual question like "Qui a tué Lee Harvey Oswald ?". The question type is the first information that we retrieve from the question analysis. It corresponds to the question discontinuous syntactic form. We defined twelve question types (QuiVerbeGN, Definition ...). The question type will not only determine the strategy of the answer search but also makes it possible to write rules to discover other informations coming from the question (answer type, question focus).

- an answer type may be a named entity (Person, Location-State, Location-City, Organization ...) or a numerical entity (Date, Length, Weight, Financial-Amount ...) or a semantic type (ie the answer may be an hyponym of this semantic type). The answer type helps locating the answer in the sentence.
- a question focus corresponds to a noun phrase that is likely to be present in the answer. The question focus serves as possible criterion for the sentence selection and also helps to the answer extraction.

The rules to find the question focus, the question type and the answer type were written from syntactic criteria and semantic knowledge extracted from the EuroWordNet thesaurus. Indeed, starting from the TreeTagger tool, we built rules making it possible to extract the question's chunks : noun phrase, etc. With the help of the EuroWordnet thesaurus, we built lists of words which are hyponyms of some predefined words which are considered as categories. For exemple, president, singer, director ... are hyponyms of Person. These lists enable us to identify for certain question type the answer type. For example, for the question "Quel est le premier ministre de la France ?" (answer type : QuelEtreGN), the word "ministre" (head of the noun phrase : "premier ministre") makes it possible to determine that the answer type must be Person.

For example, if the question is : "Qui a construit le Reichstag à Berlin ?", the analysis of this question is :

1. Question type : `QUI`
2. Answer type : `PERSON`, `ORGANIZATION`
3. Focus : `Reichstag`
4. Chunks segmentation :
   `<GN> Qui <GN> <NV> a construit </NV>`
   `</GN> le Reichstag </GN> <GP>` à `Berlin </GP>` ?
   (GN : nominal group, NV : verbal group, GP : prepositional group)
5. Verb : `construire`
6. Proper nouns : `Berlin, Reichstag`

We evaluated the question analysis by calculating, for each extracted information, the percentage of correct information.

Table 1. Evaluation of the question analysis module

Information	Percentage
Question type	99.0
Answer type	74.0
Verb	83.5
Focus	74.5

For each information, the rate of correct information is satisfactory, since higher than 74 %. Mistakes can be generated by named entity lexicons, which may be incomplete, by robust parser mistakes or by incomplete rules for some question types.

4 Sentence extraction module

The goal of this module is to extract from the journalistic corpora the most relevant sentences which answer to the question (ie, the sentences which might contain the answer). Firstly, the corpora are proceeded and marked with XML annotation in order to locate the beginning and the end of the article and of the sentences. The corpora are then annotated with part-of-speech and lemma by using the TreeTagger tool.

Then, the corpora have been indexed by Lucene search engine [11]. The indexing unit used is the sentence. For each question, we then build a Lucene request according to the data generated by the question analysis step. The request is built according to a combination of some elements linked with the "or" boolean operator. The elements are: question focus, named entities, principal verbs, common nouns, adjectives, numerical entities.

For a particular request, the sentence extraction module provides a sorted sentences list which answer to the request. The sort criterion is a confidence coefficient associated with each sentence in the list. It has been determined according to the number and the category of the question elements which are found in sentences. For example, if the question focus belongs to a sentence, the confidence coefficient of this sentence is high, because the question focus is very important for the next step of the process which is the answer extraction step. By experiments, we have defined the weight of each category, they are given in table 4. The confidence coefficient is computed by summing all the weights linked to the elements found in a selected sentence. It is then normalized. The confidence coefficient belongs to the value interval $[0, 1]$. When the sentence extraction module stops, only the 70 sentences with the highest confidence coefficient are kept.

Table 2. Weight associated with question elements

Element category	Weight
question focus	40
named entities	40
common noun	10
principal verb	15
cardinal number	10
adjective	10

After the CLEF 2005 evaluation campaign, we have studied the position of the first sentences, belonging to the list of return sentences, which contain the right answer (we except the queries whose answer was NIL).

Table 3. Sentence extraction process evaluation

Sentence position	Percentage of present answer
first sentence	40.9
2-5 sentence	22.7
6-10 sentence	9.4
+10 sentence	9.4
no sentence	17.6

As conclusion, we argue that (for queries whose answers are not NIL) 50% of them are available in the 5 first of the result set. This seems to be a satisfactory result. But, do we have so good results because of the strategy used to build the CLEF 2005 queries? Indeed, answers are often situated in sentences which contain the same words as those used for the queries.

Before this evaluation campaign, we planned to use semantic information in order to improve the sentence extraction process. But after these satisfactory obtained results, we doubt of the systematical use of semantics for improving this process. Indeed, the systematical use of semantics leads possibly to have more noise in the results. We now are working in this direction in order to determine, in which case the use of semantics brings noise in the result and in which case semantics helps to determine sentences which contain the right answer. In this aim, we are studying the contribution of topics signatures techniques (we present this technique at the end of this article).

For the next campaign, we plan to study more in details, the elements which would constitute the Lucene requests. The results would be also improved if we took into account the noun phrases in the requests (for example “tour eiffel” or “Michel Bon”). For this evaluation, in the case of the second noun phrase, the process provides the sentence: “Les ingrats en seront pour leurs frais : Michel Huet va ici jusqu’ á décerner , preuves á l’ appui la présence de plusieurs espéces de lichens sur les pierres de Notre-Dame, un brevet de bonne conduite á l’ atmosph ére parisienne !”. However, the process retrieves separately the named entity “Michel” and the adjective “Bon”. This sentence is not an answer to the request, but this error comes from the no use of noun phrase as request element.

Finally, the results would also be improved, if this module did not only provide sentences as results but passages (ie a set of sentence). For certain question, we could then use a reference tools in order to find the answer to the question.

5 Answer extraction module

We have developped two strategies to extract the answer from the queries:

- When the answer type was been determined by the question analysis step, the process extracts, from the list of sentences provided by the previous step, the named entities or the numerical entities closest to the question focus (if this last is detected). Indeed, the answer is often situated close to the question focus. For locating named entities, NEMESIS tool [6] is used. It was developed in our research team. Nemesis is a french proper name recognizer for large-scale information extraction, whose specifications have been elaborated through corpus investigation both in terms of referential categories and graphical structures. The graphical criteria are used to identify proper names and the referential classification to categorize them. The system is a classical one: it is rule-based and uses specialized lexicons without any linguistic preprocessing. Its originality consists on a modular architecture which includes a learning process.
- When the answer type is not available, the process uses syntactical patterns in order to extract answers. Indeed, according to the question type, certain syntactical patterns can be employed. These patterns were built by taking into account the presence of the question focus and its place compared to the answer. For example, for the question “Qu’est ce que les FARC ?” whose category is definitional, the system uses the following pattern : GNRep (GNFocus). We give here an example of sentence where the system applies the previous pattern in order to find an answer: “Les deux groupes de guérilla toujours actifs , ¡GNRep¿ les Forces armées révolutionnaires de Colombie ¡/GNRep¿ (¡GNFocus¿ FARC ¡/GNFocus¿) et , dans une moindre mesure , l’ Armée de libération nationale (ELN , castriste) exécutent des paysans accusés d’ être des informateurs ou des guérilleros ayant déposé les armes.”. According to the pattern, the system extracts the answer (“Les Forces armées révolutionnaires de Colombie”).

After our system evaluation for the french monolingual task, we have obtained the following results:

Table 4. Evaluation of the question answering system

Answer type	Number of right answer
Numerical entity	7
Named entity	14
NIL	3
Definition	3
Other queries	2

The results are not satisfactory, because we only recover 29 correct answers. After analysing the results, we see that the majority of correct answers correspond to queries whose answers are a named entity or a numerical entity. Moreover, as seen in paragraph 3, for the question analysis step, 26% of the answer types for definitional questions were incorrect. We can then easily improve the process for these question types. On the other hand, the use of syntactic patterns is not satisfactory for the system for several reasons:

- the chunk analyser is not complete;
- the syntactic patterns were built according to learning techniques. The process has been trained on a restricted set of questions(100) coming from an old evaluation campaign. Then, certain question types were not linked to their own answer extraction patterns;
- we do not use semantic pattern in order to extract answer.

6 Conclusion and Prospects

The system has not obtained a high number of correct answers, but it was its first evaluation campaign. The interest to have participated is to highlight changes which can easily improve the system results. 25 questions among the proposed questions were particular definitional questions. For these questions, the answer was the meaning of an abbreviation. If we used an abbreviation recognizer, we would be able to answer to 19 of these question types (the 6 others are abbreviations coming from a foreign language and whose meaning is given in french in the retrieved sentences). The syntactic patterns, used in the answer extraction module, do not cover the totality of the question types set. Indeed, the learning process was performed on a small sample of questions (100) coming from old evaluation campaigns. Several types of question were not present in the sample. The major improvement is to perform the learning process on an other more complete sample and also to add new syntactic patterns manually.

In perspective, we study the use of semantics in order to improve the query answering system by the use of semantics based techniques .

The Wordnet thesaurus is often used for semantic processing of textual data. One of the principal difficulties is to determine ”the right sense” for polysemous terms. In spite of a weak rate of polysemia in Wordnet (approximately 18%), in practice, the need to disambiguate terms is frequent (78% of the terms are polysemous in a corpus like SemCor) [7]. Methods to disambiguate a term are numerous [9]. These methods, although powerful, appear limited when the context is small or the structure is weak. A method seems interesting in these situations: the use of the topic signatures [8].

This method, like others [10], uses the Web as a corpus. The first step to disambiguate a term is to build a corpora of HTML documents associated with each sense of this polysemous term. From these corpora, sets of terms are associated with all the different senses of the polysemous term. Then, either by using the X2 function [8] or by using the tf.idf measure [2], the sets are reduced according to the terms which make it possible to discriminate the senses of the polysemous term: the topic signatures.

From these topic signatures, it is then possible to determine the sense of a term according to its context. Regarding QA systems, we think that topic signatures make it possible to improve the process at various levels. Firstly, during the analysis of the question, it makes it possible to improve the disambiguation of the terms. Indeed, the very poor context of the question does not always make it possible to decide which is the correct sense. Secondly, the set of terms associated with a given sense makes it possible to improve the request provided to the search engine and also to optimize the identification of the passages where the answer might be found.

References

1. Vossen P. : “EuroWordNet: A Multilingual Database with Lexical Semantic”, editor Networks Piek Vossen, university of Amsterdam, 1998.
2. Agirre E., Lopez de Lacalle O. : “topic signature for all WordNet nominal senses”, Publicly available, LREC 2004.
3. Schmid H. “Improvements in Part-of-Speech Tagging with an Application To German”. In Armstrong, S., Chuch, K. W., Isabelle P., Tzoukermann, E. & Yarowski, D. (Eds.), NaturalLanguage Processing Using Very Large Corpora, Dordrecht: Kluwer Academic Publisher.1999
4. Fellbaum, C. (1998). WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.
5. Monceaux L. : “Adaptation du niveau d’analyse des interventions dans un dialogue - application à un système de question - réponse”, These en informatique, Paris Sud, ORSAY, LIMSI (2003)
6. Fourour, N. : “Identification et catégorisation automatiques des entités nommées dans les textes franais”, These en informatique, Nantes, LINA(2004)
7. De Loupy C. : “Evaluation des taux de synonymie et de polysḿie dans un texte”, TALN2002, Nancy, pp.225-234
8. Agirre E., Ansa 0., Hovy E., Martinez D. : “Enriching very large ontologies using WWW.”, Proceeding of the Ontology Learning Workshop ECAI 2000
9. Ide N., Véronis J.: Word Sense Disambiguation: The State of the Art. Computational Linguistics, 1998, 24(1)
10. Mihalcea R., Moldovan D.I. : “An Automatic Method for Generating Sense Tagged Corpora.”, Proceeding of AAAI’99, 1999, pp. 461-466
11. Lucene search engine: http://lucene.apache.org/java/docs/