QA@L^{2}F, first steps at QA@CLEF

Ana Mendes, Luísa Coheur, Nuno J. Mamede
Ricardo Ribeiro, Fernando Batista, David Martins de Matos

L^{2}F/INESC-ID Lisboa
Rua Alves Redol, 9, 1000-029 Lisboa, Portugal
`qa-clef@l2f.inesc-id.pt`
`http://www.l2f.inesc-id.pt`

Abstract. This paper presents QA@L^{2}F, the question-answering system developed at L^{2}F, INESC-ID. QA@L^{2}F follows different strategies according with the question type, and relies strongly on named entity recognition and on the pre-detection of linguistic patterns. Each question type is mapped into a single strategy; however, if no answer is found, the system proceeds and tries to find an answer using one of the other strategies.

1 Introduction

In this paper we present QA@L^{2}F, the question-answering system from L^{2}F, INESC-ID, as well as the results obtained at CLEF 2007.

In general terms, we can say that QA@L^{2}F executes the following tasks:

- Information Extraction: information sources are processed, in order to extract potentially relevant information (such as named entities or relations between concepts), which is stored into a database;
- Question Interpretation: question is interpreted and mapped into an SQL query;
- Answer Finding: according with the question type, different strategies are followed in order to find the answer.

Considering information extraction, if a QA system focus on a particular domain or if the system is going to be used in an evaluation where the information sources are known, it makes sense to process all that information off-line, in order to get potentially relevant information. Thus, for the CLEF competition, QA@L^{2}F pre-process the available corpora and gets structured information (such as named entities or noun phrases) that might be the answer to potential questions. This task is performed by many systems, as for instance Senso [14, 17].

Either to extract information or to interpret the question, some systems use natural language processing techniques [5, 8]; some perform named entity recognition and co-reference resolution. Also, many systems profit from thesaurus [5, 2, 8, 9] or ontologies [14, 17]. Internet may also be used as a resource [6, 4].

In what concerns QA@L^{2}F, it profits from a Natural Language Processing (NLP) chain, which performs morpho-syntactic analysis, named entity recognition and shallow semantic analysis based on the named entities [10, 16]. This NLP chain uses the following tools:

- Palavroso [11], responsible for the morphological analysis and MARv [15] for its disambiguation;
- Rudrico (an improved version of PAsMo [13]), which not only recognize multi-word terms and collapse them into single tokens, but also splits tokens;
- XIP [1], which returns the input organized in chunks and connected by dependency relations.

This chain is used both in the information extraction step and in question interpretation.

In order to find the answer, systems such as INAOE [7] focus on the question type and follow different strategies according to it. QA@L2F also applies different strategies depending on the question type. However, if no answer is found, the system relaxes and tries to find an answer using one of the other strategies. Typically, several snippets are answer candidates and the QA system has to choose one of them. Although there are systems such as QUASAR [3] that combine frequency and the confidence given to both answer candidate and text passage in which the answer can be found, many systems choose the most frequent of all possible answers [18]. A confidence level is used by QA@L^{2}F in one of its strategies; all the others only take frequency into consideration.

This paper is organized as follows: section 2 focus on the information extraction step; section 3 details the question interpretation; section 4 describes the different methods used to find the answer; section 5 presents and discusses the evaluation results; finally, section 6 concludes and points to future work.

2 Information Extraction

In order to extract information from newspaper corpora, a morpho-syntactic analysis is used to identify named entities, such as PEOPLE, which refer to person's names, CULTURE, to pieces of art, and TITLE, to person's professions and titles. With this information, as well as with a set of manually built linguistic patterns, relations between concepts are captured by the same NLP-chain, and stored into a database (from now on, the "relation-concepts" database). Every named entity recognized is also stored into a database (from now on, the "named entities" database).[1]

For instance, consider the sentence *"Land and Freedom, de Ken Loach, evocação da Guerra Civil Espanhola"* (*"Land and Freedom, by Ken Loach, an evocation of the Spanish Civil War").* In this piece of information might lay the answer to the question *"Who directed Land and Freedom?"*. Therefore, by using linguistic patterns, the entry in the relation-concepts database from table 2 is built.

[1] As it should be clear, in both situations, the reference to the text snippet holding those relations/entities is also kept.

CULTURE

id	culture	author	confidence	count
1	Land and Freedom	Ken Loach	99	4

Table 1. Entry representing the relation between *Ken Loach* and *Land and Freedom*.

It should be noticed that these relation-concepts tables have information concerning the confidence given to that relation. It depends on the confidence level given to the linguistic patterns, which are assigned manually. Notice also, that "count" represents the frequency of this relation in the processed corpus.

In what concerns Wikipedia, QA@L^{2}F used the WikiXML collection provided by the Information and Language Processing Systems group at the Informatics Institute, University of Amsterdam, as well as its database structure[2]. A new table containing only the XML article nodes from every Wikipedia page, with no linguistic processing, was also created. The aim was to answer definition questions.

3 Question Interpretation

In QA@L^{2}F, the question interpretation step is responsible for the transformation of the question into a SQL query.

The question is processed by: a) the NLP chain, which recovers the type of the question, as well as other information considered relevant (such as named entities and the question focus); b) a SQL generator.

Considering the question *"Quem é Boaventura Kloppenburg?"* (*"Who is Boaventura Kloppenburg?"*), after the NLP chain, both the type (WHO_PEOPLE) and the focus (*Boaventura Kloppenburg*) of the question are identified:

```
<DEPENDENCY name="WHO_PEOPLE">
<PARAMETER ind="0" num="11" word="Boaventura Kloppenburg"/>
</DEPENDENCY>
```

The SQL generator comprises the steps shown in Figure 1.

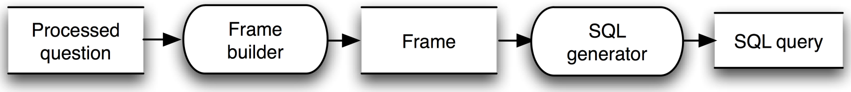

Fig. 1. SQL generation.

The frame builder is responsible for chosing:

[2] http://ilps.science.uva.nl/WikiXML/

- the answer extraction script to be called next (depending on the type of the question);
- the question focus;
- all the named entities identified in the question.

The SQL generation is performed by a set of scripts that maps the frames into a SQL query.

Considering the previous example, the following frame is built:

```
SCRIPT    script-who-people.pl
TARGET    "Boaventura Kloppenburg"
ENTITIES  "Boaventura Kloppenburg" PEOPLE
```

This frame is then mapped into the following MySQL query, that will *possibly* retrieve the question's answer:

```
SELECT   title, confidence, count
FROM     FACT_PEOPLE
WHERE    name="Boaventura Kloppenburg"
GROUP BY confidence DESC, count DESC
```

The "relation-concepts" database is queried and every title (or profession) connected with *Boaventura Kloppenburg* is retrieved, in descendant order of confidence and frequency.

4 Answer Finding

QA@L^{2}F has a set of answer finding strategies. From within this set, the system has a prefered one to be applied on each question, depending on its type. The system expects this strategy to give the correct answer.

As an example, if the submitted question can be answered directly using the "relation-concepts" database, the system will just query that database. If not, the system adopts the following strategies, depending on the type of the question:

- *Linguistic Reordering*: the answer is searched in the wikipedia, after a reordering of some question elements;
- *Named Entities Matching*: the answer is searched in the named entities database;
- *Brute Force plus NLP*: some text snippets are chosen and processed in runtime; the obtained information provides QA@L^{2}F a last chance to find an answer.

After detecting the type of question, one of these strategies is followed. If no answer is found, the system tries to answer it by using other strategy.

Using a method that allows it to jump to another strategy if the first one applied did not succeed, implicitly makes the system relax its constraints: it applies a strategy, even if it is not the one in which it relies the most to use on that question.

4.1 Linguistic Reordering

This strategy is used mainly for answering definition questions, like *Quem foi Pirro?* (*Who was Pirro?*) and *O que é a Igreja Maronita?* (*What is the Maronite Church?*), or list questions, like *Diga uma escritora sarda.* (*Mention a sardinian writer.*).

QA@L^{2}F uses Wikipedia in order to answer that group of questions. Firstly, the question interpretation step recovers the question focus (*Pirro*, *Igreja Marronita* and *escritora sarda*, considering the above examples). Then, it performs a search over the articles and applies the patterns inferred by the question structure to find the answer.

For definition questions, patterns are of the form: *question focus* plus the inflected verb *to be*. For instance, *Pirro foi...* (*Pirro was...*) or *Maronite Chuch é...*(*Maronite Church was...*). On the other hand, for list questions, those patterns are of the form: the inflected *to be* plus the *question focus*. For instance, *...é uma escritora sarda*(*...is a sardinian writer*).

This strategy is also used on questions for which the system could not find an answer using the linguistic patterns matching technique. Consider, for instance, the question *Quem foi Ésquilo?* (*"Who was Aeschylus?"*). The relation between *Ésquilo* and his title was not captured using linguistic patterns. Thus, the system searched on Wikipedia for the page having *Ésquilo* as title. The information about *Ésquilo*'s definition, a tragic greek poet, was found by processing the first line of this Wikipedia article page and, finally, returned as the question's answer.

4.2 Named Entities matching

This method queries the named entities database. A set of text snippets containing the named entities of the question is retrieved.

For instance, during the question interpretation of *Quem sucedeu a Augusto?* (*Who came after Augustus?*), the following frame was built:

```
TARGET EMPTY
ENTIDADES  "Augusto " PEOPLE
AUXILIARES "sucedeu" ACTION "a Augusto"
```

With this information, QA@L^{2}F searches on the database for snippets containing the named entity of type PEOPLE *Augusto* and the words *sucedeu* and *a Augusto*. For these last two, since they are not classified as named entities, the system performs a full-text query against the text snippets. The system gathers all the named-entities of types PEOPLE and PROPER (NAME) on those snippets, classifies them by order of frequency and returns the most frequent. Due to the fact that the system discards every candidate answer matching any word in the built frame, the named-entity *Augusto* is not chosen as the final answer.

4.3 Brute-Force plus NLP

If none of the previously described strategies finds an answer, the system performs a full-text query against the raw text snippets database, returning the top ten best qualified snippets. Those snippets are processed by the NLP chain and the most frequent concept matching the wanted answer type is returned.

It should be noticed that this strategy is also used because we did not apply the information extraction module over the entire corpora. As so, although all the information is in the database, sometimes it is just in the form of a text snipped, without any processing. This technique allow us to extract information in run-time from paragraphs considered relevant.

4.4 Choosing the Answer

The system uses two main approaches in order to retrieve the final answer, depending on the strategy followed.

If the chosen strategy is either the linguistic patterns matching or the linguistic reordering, the system simply returns the answers found and takes in consideration the confidence and count attributes of each table (if they exist).

On the other hand, if the chosen strategy is either the named-entity recognition or the brute-force plus NLP, the answer extraction step depends on the type of the question. Having in mind that we are dealing with large corpora (564MB of newspaper text, both in European Portuguese and Brazilian Portuguese, as well as the Wikipedia pages found in the version of November, 2006), the system assumes that the correct answer is repeated on more than one text snippet. With this assumption, QA@L^{2}F returns the most frequent named entity that matches the type of the question.

5 Evaluation

QA@L^{2}F participated and was evaluated at CLEF for Portuguese as the query and target language. Table 2 presents the obtained results.

Right	Wrong	ineXact	Unsupported	Total	Accuracy (%)
28	166	4	2	200	28/200 = 14%

Table 2. QA@L^{2}F results at CLEF 2007.

Considering the correct answers:

- 11 were NIL;
- 3 followed the direct query of the "‘relation-concepts” database;
- 14 followed the linguistic reordering;

– from these 17, 2 used the relaxing mechanism.

It should be noticed that only 114 questions were interpreted (anaphora, ellipsis and some question types were not addressed).

Considering the ineXact answers, QA@L^{2}F answered only the identified named entity, resulting into a ineXact answer. Nevertheless, it is difficult to be objective in deciding what should be the exact answer.

For instance, in the question *"Quem é George Vassiliou?"* (*"Who is George Vassiliou?"*) it is obvious that the answer *"presidente de Chipre"* (*"Cypriot president"*) is incomplete, as he was *"presidente de Chipre entre 88 e 93"* (*"Cypriot president between 88 and 93"*). However, being given the following paragraph – *"...norueguês, Henrik Ibsen, dramaturgo que escreveu Peer Gynt."*(*"...norwegian, Henrik Ibsen, dramaturge that wrote Peer Gynt"*) – it is not so obvious what should be the right answer to *"Quem foi Henrik Ibsen?"* (*"Who was Henrik Ibsen?"*").

If *"dramaturgo"* is incomplete, is *"dramaturgo norueguês"* enough? Or the right answer should be *"dramaturgo norueguês que escreveu Peer Gynt"*? It is difficult to decide.

Details about the evaluation can be found in [12].

6 Conclusions and future work

This paper presents QA@L^{2}F first steps. The system follows different strategies according to the type of the submitted question and bases its performance on named entity recognition; if no answer is found, the system relaxes and tries to find the answer using another strategy.

Many improvements are yet to be done to QA@L^{2}F. The improvement of all of the steps/techniques described in this paper are already scheduled, however the introduction of new strategies is also considered a goal.

Besides the current existence of a linguistic patterns matching approach, we would like to explore a syntactic pattern matching strategy, using patterns at the syntatic level.

We also would like to explore in detail Wikipedia's standard structure (namely how it stores birh and death days and places, for instance), as it allows an easy retrieval of miscellaneous information.

References

1. Salah A´it-Mokhtar, Jean-Pierre Chanod, and Claude Roux. A multi-input dependency parser. In *Proceedings of the Seventh IWPT (International Workshop on Parsing Technologies)*, Beijing, China, October 2001.
2. Carlos Amaral, Adán Cassan, Helena Figueira, André Martins, Afonso Mendes, Pedro Mendes, Cláudia Pinto, and Daniel Vidal. Priberam's question answering system in qa@clef 2007. *Working Notes for the CLEF 2007 Workshop*, 2007.
3. Davide Buscaldi, Yassine Benajiba, Paolo Rosso, and Emilio Sanchis. The UPV at QA@CLEF 2007. *Working Notes for the CLEF 2007 Workshop*, 2007.

4. Luís Miguel Cabral, Luís Fernando Costa, and Diana Santos. Esfinge at CLEF 2007: First steps in a multiple question and multiple answer approach. *Working Notes for the CLEF 2007 Workshop*, 2007.
5. Adán Cassan, Helena Figueira, André Martins, Afonso Mendes, Pedro Mendes, Cláudia Pinto, and Daniel Vidal. Priberam's question answering system in a cross-language environment. *Working Notes for the CLEF 2006 Workshop*, 2006.
6. Luís Costa. Esfinge - a modular question answering system for portuguese. *Working Notes for the CLEF 2006 Workshop*, 2006.
7. Antonio Juárez-Gonzalez, Alberto Téllez-Valero, Claudia Denicia-Carral, Manuel Montes y Gómez, and Luis Villase nor Pineda. INAOE at CLEF 2006: Experiments in Spanish Question Answering. *Working Notes for the CLEF 2006 Workshop*, 2006.
8. Dominique Laurent, Patrick Séguéla, and Sophie Nègre. Cross Lingual Question Answer using QRISTAL for CLEF 2006. *Working Notes for the CLEF 2006 Workshop*, 2006.
9. Dominique Laurent, Patrick Séguéla, and Sophie Nègre. Cross Lingual Question Answering using QRISTAL for CLEF 2007. *Working Notes for the CLEF 2007 Workshop*, 2007.
10. João Loureiro. NER - Reconhecimento de Pessoas, Organizações e Tempo. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Portugal, 2007.
11. José Carlos Medeiros. Análise morfológica e correcção ortográfica do português. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Portugal, 1995.
12. Ana Mendes. Clefomania, QA@L2F: Primeiros Passos. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Portugal, 2007.
13. Joana Paulo Pardal and Nuno J. Mamede. Terms spotting with linguistics and statistics. In *Proceedings of the international workshop "Taller de Herramientas y Recursos Linguísticos para el Espanõl y el Portugués", IX Iberoamerican Conference on Artificial Intelligence (IBERAMIA 2004)*, pages 298–304, November 2004.
14. Paulo Quaresma and Irene Rodrigues. A logic programming based approach to the QA@CLEF05 track. *Working Notes for the CLEF 2005 Workshop*, 2005.
15. Ricardo Ribeiro, Nuno J. Mamede, and Isabel Trancoso. Using Morphossyntactic Information in TTS Systems: comparing strategies for European Portuguese. In *Computational Processing of the Portuguese Language: 6th International Workshop, PROPOR 2003, Faro, Portugal, June 26-27, 2003. Proceedings*, volume 2721 of *Lecture Notes in Computer Science*. Springer, 2003.
16. Luis Romão. NER - Reconhecimento de Locais e Eventos. Master's thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa, Portugal, 2007.
17. José Saias and Paulo Quaresma. The Senso Question Answering Approach to Portuguese QA@CLEF-2007. *Working Notes for the CLEF 2007 Workshop*, 2007.
18. Luís Sarmento. Hunting answers with RAPOSA (FOX). *Working Notes for the CLEF 2006 Workshop*, 2006.