Question Interpretation in QA@L^{2}F

Luísa Coheur, Ana Mendes, João Guimarães
Nuno J. Mamede, Ricardo Ribeiro

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

Abstract. The Question Interpretation module of QA@L^{2}F, the question-answering system from L^{2}F/INESC-ID, is thoroughly described in this paper, as well as the frame formalism[1] it employs. Moreover, the anaphora resolution process introduced this year, based on frames manipulation, is detailed.
The overall results QA@L^{2}F achieved at the CLEF competition and a brief overview on the system's evolution throughout the 2 years of joint evaluation are presented. The results of an evaluation to the QI module alone are also detailed here.

1 Introduction

QA@L^{2}F is the question-answering (QA) system from L^{2}F/INESC-ID, that participated in 2007 and 2008 in the monolingual QA task of CLEF [2]. To answer questions, the system follows three main steps:

- Corpus Pre-Processing: as in many QA systems, like Senso [3], information sources are partly processed in order to extract potentially relevant information, like named entities and relations between concepts. The latter information represents possible answers to questions and is stored in a database;
- Question Interpretation (QI): the question is analysed and transformed into a frame, which is mapped into an SQL query or used to search relevant snippets;
- Answer Extraction: each question type is mapped into a single strategy. As a result, depending on the question type, different strategies are used to find the answer. However, if no answer is found, the system proceeds and tries to find an answer using alternative strategies. Details on these strategies can be found in [2].

This paper focus on the QI module of QA@L^{2}F, and is organized as follows: section 2 presents related work; section 3 describes the QI module, including the anaphora resolution process; section 4 shows, discusses and compares evaluation results; finally, section 5 concludes and points to future work.

[1] As in [1], we call 'frame' to a set of slot-value pairs; we call 'frame element' to each slot-value pair.

2 Related Work

Work in questions' processing can be split into two main tasks: question classification and QI. Question classification aims at mapping different question types into proper semantic categories. For instance, [4] proposes a method to classify *what-type* questions based on head noun tagging. QI goal is the conversion of natural language questions into structured information, more suitable for computer processing. Typically, these structures are logical forms or frames, but they can also be questions in natural language that the computer already understands. Clearly, question classification has an important role in QI. Although some systems implement hybrid approaches, involved techniques can be classified as: a) basic QI; b) statistical QI; c) linguistically-motivated QI.

Basic QI includes keyword detection, pattern matching and the use of simple algorithms capable of associating new input to already understood utterances. Although not focused on QA, a classical example that perfectly illustrates a system based on pattern matching, is the well-known ELIZA [5], invented in the early 1960's, aiming at emulating a psychologist.

In what concerns statistical QI, there are several techniques being explored, coming some of them from the Machine Learning framework. The main problem of using statistical techniques in QI is the small size of the potential training data. An example of a work that applies statistical methods to little training data is the one presented in [1], where four different techniques are applied to a training set (not only questions) constituted by 477 sentence/frame pairs. Results from this evaluation ranged from a 0.75 F-score to a 0.83 F-score. It should be noticed, however, that these results derived from the fact that the domain was limited and it was possible to replace each entity of the domain by its correspondent class name. In an open QA system is not obvious that these techniques would obtain similar results.

Finally, linguistically-motivated QI use some level of linguistic information. Some systems implementing this paradigm base their performance on a syntax/semantics interface, where each syntactic rule is associated with a semantic rule and logical forms are generated in a bottom-up, compositional process. Variations of this approach are followed by several systems. Two of the most referenced books in Natural Language Processing, that is [6] and [7], depict this approach. Also, last year, QA@L^2F [2] followed these lines, although a slightly different (but also common) linguistically-motivated technique was used: a semantic module was operating over a dependency structure, obtained after a cascaded syntactic/semantic analysis.

Due to a strong dependency between the semantic and the syntactic analysis, that brought many problems to the semantic analysis, this year QA@L^2F follows a different strategy that combines a) and c) approaches. On the one hand, question classification is based on a sophisticated pattern matching, that uses morpho-syntactic patterns. On the other hand, the module profits from a named entity recognizer based on a deep linguistic analysis of the question, in order to identify relevant entities (people, titles, locations, dates, etc.). This information is merged in order to create a frame. This process is detailed in the next section.

3 Question Interpretation

The QI module of QA@L^{2}F module involves the following steps:

- morphological analysis, performed by Palavroso [8] and MARv [9];
- creation of intermediary frames (pre-frame), representing relevant information extracted from the question. This step is performed by RuDriCo (an improved version of PAsMo [10]), which is a rule-based tool that recognizes multi-word term and collapses them into single tokens; it can also split tokens. RuDricCo's rules are patterns that match against labeled text, being RuDriCo the tool responsible for the sophisticated pattern matching that uses morpho-syntactic patterns, as mentioned before;
- named entity recognition, performed by XIP [11], a tool that returns the input organized into chunks, connected by dependency relations, and also identifies and classifies the named entities in the input. As mentioned before, XIP bases the named entity recognition in a deep linguistic analysis;
- frames creation, in which information from the pre-frame is merged with the information returned by the named entity recognizer.

Figure 1 depicts the entire question interpretation process used in QA@L^{2}F.

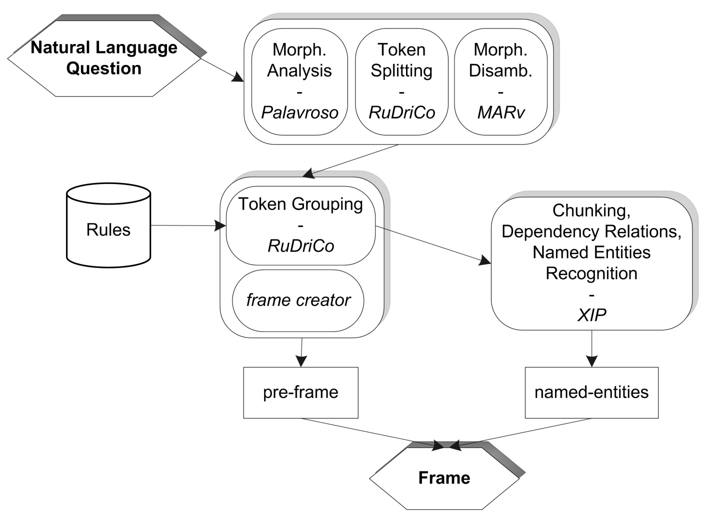

Fig. 1. Question interpretation in QA@L^{2}F.

The following example shows a RuDriCo rule, which is able to capture questions such as *Quando nasceu Thomas Mann* (*When was Thomas Mann born?*), and responsible for a pre-frame creation.

```
S1 ['quando','CAT'/C1]
S2 ['ser','CAT'/C2]? 'que' [L3,'CAT'/C3]?
S4 [L4,'CAT'/'nascer']
S6 [L6,'CAT'/'noun']
S7 [L7,'HMM'/'true']*
S10 ['?','CAT'/C10] -->
   S1 ['onde', 'CAT'/C1, 'type'/'onde_verb']
   S4 [L2,'CAT'/'verb']
   S6@+S7@* [L6@+L7@*, 'type'/'target'].
```

The left side (before the arrow) of this rule matches the question; the right side outputs the frame elements that constitute the pre-frame. XIP outputs the named entities. Both results are merged into a final frame. In the next section, the frames formalism used in QA@L^{2}F is described.

3.1 Frames

Each frame in QA@L^{2}F consists of the following elements:

- the question type: a string that identifies the script to be called;
- the question target: a string that represents the question main entity;
- named entities: a set of strings representing the entities identified by the named entity recognizer;
- auxiliaries: a set of strings constituted by auxiliary (and optional) elements from the question, like the target-type, main verbs, adjectives and adverbs.

Considering the previous question *Quando nasceu Thomas Mann?*, its corresponding frame is:

Frame
when/script-wiki-target.pl
target="thomas mann"
entities people="thomas mann"
auxiliaries verb="nasceu"

The question type is identified by the script `when/script-wiki-target.pl`, the question target is `thomas mann` which is also identified by the named entity recognizer as *people*; the auxiliaries' set is constituted solely by the verb `nasceu`.

The obtained script is then called and uses the other frame elements either to build the SQL query or to obtain the snippets that may contain the answer. Details about this step can be found in [2].

3.2 Anaphora resolution

The capability of handling anaphora plays an important role along the entire pipeline of QA systems. It can have much impact in the performance of its compounding modules: on one hand, the benefits of analyzing, identifying and solving anaphoric references during the corpus pre-processing stage are shown in the studies driven by [12]; on the other hand, during the question interpretation step, anaphora resolution is also applied in order to deal with follow-up questions, as pointed by [13].

Like other systems that already deal with this linguistic phenomenon (see, for instance, [14] and [3]), QA@L^{2}F integrates a module for pronominal anaphora resolution for follow-up questions. In addition, ellipsis, being a special case of anaphora, is also addressed in this module.

Anaphora resolution is based on insertions/replacements over the frame elements of anaphoric questions. The frame associated with the reference question provides the frame elements to be used in these insertions/replacements. For instance, in pronominal anaphora, the target pronoun is replaced by the target of the reference question. In order to illustrate this procedure, consider the next group of questions [2]:

1. *Onde nasceu a Florbela Espanca?* (*Where was Florbela Espanca born?*)
2. *Quando?* (*When?*)
3. *Onde morreu ela?* (*Where did she die?*)

The following frame was generated by the reference question:

Reference
where/script-wiki-target.pl
target="florbela espanca"
entities people="florbela espanca"
auxiliaries verb="nasceu"

The manipulated frames for each of the follow-up questions are shown next [3]:

Elliptic question	Pronominal question
when/script-wiki-target.pl	***where/script-wiki-target.pl***
target="florbela espanca"	target="florbela espanca"
entities people="florbela espanca"	entities people="florbela espanca"
auxiliares verb="nasceu"	***auxiliares verb="morreu"***

This module bases its actions on the assumption that only the information introduced by the reference question can be used in anaphora resolution. However, this is not always the case, since follow-up questions can also provide information to further questions. Developments on this module are, thus, still required.

[2] We will call "reference question" to the first question, "elliptic question" to the second question and "pronominal question" to the last one.

[3] Frame elements that do not result from insertions/replacements from the frame associated with the reference question are displayed in bold italic.

4 Evaluation

QA@L^{2}F was evaluated at CLEF, using Portuguese as source and target languages. This section presents the QI step evaluation as well as the system final results.

In what concerns the QI step, frames were generated and then manually evaluated in terms of its correctness according to the expected frame. The results are the following:

```
Total: 200 questions

Right: 113
Wrong:  87
     Total fail:      14
     Partially wrong: 73
           Wrong script:      27
           Wrong target:      50
           Wrong entities:     7
           Wrong auxiliaries: 50
```

As it can be seen, from the 200 questions that constituted the test set, the QI module succeeded in creating the correct frame in 56,5% of the cases. In 14 of the 87 wrong frames, the module completely failed to create the frame. The other items represent which of the frame components were wrongly identified.

Considering only anaphoric questions, 13 of the 52 follow-up questions where mapped into the correct frame, resulting in an accuracy of 25%. It should be noticed that in these 13 frames, 4 were incorrect due to errors occurred in the generation of the reference frame.

Table 1 shows the final results. The system had better overall results this year: 20% of correct answers, against 14% last year. However, the number of wrong answers continues high (150), although it decreased from 166 since 2007.

Table 1. QA@L^{2}F results at CLEF 2008.

Right	Wrong	ineXact	Unsupported	Accuracy over the FIRST answer (%)
40	150	5	5	40/200 = 20%

Table 2 shows the detailed results for each question type. Just like what happened at the competition in 2007, the system obtained this year the best results in definition questions. Also, the accuracy in factoid questions improved: 22 factoid questions were answered correctly (corresponding to 13.580% of precision), against only 8 (5.03%) in the last year. Moreover, the system answered correctly to one list question: last year no correct answers were given to any question of this type.

Table 2. QA@L^{2}F results for each question type.

Question Type	Total	Right	Wrong	ineXact	Unsupported	Accuracy (%)
Factoids	162	22	132	3	5	22/162 = 13.580%
Lists	10	1	8	1	0	1/10 = 10.0%
Definition	28	17	10	1	0	17/28 = 60.714%
Temporally Restricted	16	1	14	0	1	1/16 = 6.250%

It is worth to mention that we did not profit from the fact that the system could return 3 answers. In fact, the distribution of the 230 answers given by our system was the following: 184 single answers, 2 double answers and 14 triple answers. Moreover, several answers were extracted from Wikipedia's tables and, although the page from where they were extracted was correctly identified, they were considered unsupported.

5 Conclusions and future work

We presented the QI module of QA@L^{2}F, which uses a linguistically-motivated pattern matching system to obtain part of a frame and that profits from a named entity recognizer to build the whole frame. Moreover, anaphora is solved by manipulating frames, according to the type of the involved questions. Results about these processes were also presented, as well as the results obtained by QA@L^{2}F in QA@CLEF08.

In the near future we intend to improve the QI module, not only by expanding its rules, but also by exploring other techniques. Also, we need to evaluate the impact of each one of the different types of the errors in the system capability of obtaining a correct answer.

Although the entire system needs strong improvements, there are many small things to be done in QA@L^{2}F that can make it achieve better results. In the following we detach some of these improvements:

- Validate the answer type: 10 out of the 150 wrong answers do not have the expected type from the question. Being so, and already possessing a tool that is able to say that something is a *PERSON* or a *LOCATION* (for instance), it is not difficult to validate an answer type, as this is already retrieved from the question. This will certainly give better results, when articulated with redundancy, than only using redundancy by itself.
- Return 3 answers: as said before, only 230 answers (of which 14 triple) where returned this year;
- Improve the anaphora solver: as mentioned before, the system only solves anaphoras based on the frame constructed for the first question of a group of related questions and an anaphora can be related with any question from that group.

References

1. Bhagat, R., Leuski, A., Hovy, E.: Shallow semantic parsing despite little training data. In: Proc. ACL/SIGPARSE 9*th* Int. Workshop on Parsing Technologies. (2005)
2. Mendes, A., Coheur, L., Mamede, N.J., Ribeiro, R., Batista, F., de Matos, D.M.: QA@L2F, first steps at QA@CLEF. In: Proc. Cross-Language Evaluation Forum 2007. Lecture Notes in Computer Science, Springer (2008)
3. Amaral, C., Cassan, A., Figueira, H., Martins, A., Mendes, A., Mendes, P., Pinto, C., Vidal, D.: Priberam's Question Answering System in QA@CLEF 2007. In: Proc. Cross-Language Evaluation Forum 2007. Lecture Notes in Computer Science, Springer (2008)
4. Li, F., Zhang, X., Yuan, J., Zhu, X.: Classifying what-type questions by head noun tagging. In: Proc. 22*nd* Int. Conference on Computational Linguistics (Coling). (2008)
5. Weizenbaum, J.: ELIZA—a computer program for the study of natural language communication between man and machine. Communications of the Association for Computing Machinery **9**(1) (1965) 36–45
6. Allen, J.: Natural language understanding (2*nd* ed.). Benjamin-Cummings Publishing Co., Inc. (1995)
7. Jurafsky, D., Martin, J.H.: Speech and Language Processing (2*nd* Edition). Prentice-Hall, Inc. (2006)
8. Medeiros, J.C.: Análise Morfológica e Correcção Ortográfica do Português. Master's thesis, Instituto Superior Técnico, Univ. Técnica Lisboa, Portugal (1995)
9. Ribeiro, R., Mamede, N.J., Trancoso, I.: Using Morphossyntactic Information in TTS Systems: comparing strategies for European Portuguese. In: Proc of PROPOR, Computational Processing of the Portuguese Language: 6*th* Int. Workshop. Lecture Notes in Computer Science, Springer (2003)
10. Pardal, J.P., Mamede, N.J.: Terms Spotting with Linguistics and Statistics. In: Proc. Int. Workshop "Taller de Herramientas y Recursos Linguísticos para el Espanõl y el Portugués", IX Iberoamerican Conference on Artificial Intelligence (IBERAMIA). (2004)
11. Aït-Mokhtar, S., Chanod, J.P., Roux, C.: A Multi-Input Dependency Parser. In: Proc. 7*th* Int. Workshop on Parsing Technologies. (2001)
12. Vicedo, J.L., Ferrández, A.: Importance of pronominal anaphora resolution in question answering systems. In: ACL '00: Proc. of the 38*th* Annual Meeting of the ACL. (2000)
13. Negri, M., Kouylekov, M.O.: Who Are We Talking About? Tracking the Referent in a Question Answering Series. In: Discourse Anaphora and Anaphor Resolution Colloquium, Springer Verlag (2007) 167–178
14. Bouma, G., Kloosterman, G., Mur, J., van Noord, G., van der Plas, L., Tiedemann, J.: Question Answering with Joost at CLEF 2007. In: Proc. Cross-Language Evaluation Forum 2007. Lecture Notes in Computer Science, Springer (2007)