Question Answering on Romanian, English and French Languages

Adrian Iftene[1], Diana Trandabăţ[1, 2], Maria Husarciuc[3], Alex Moruz[1, 2]

[1] UAIC: Faculty of Computer Science, "Alexandru Ioan Cuza" University, Romania
[2] Institute of Computer Science, Romanian Academy Iasi Branch
[3] Center of Biblical-Philological Studies *Monumenta linguae Dacoromanorum*,
"Alexandru Ioan Cuza" University, Romania
{adiftene, dtrandabat, mhusarciuc, amoruz}@info.uaic.ro

Abstract. This paper describes UAIC[1]'s Question Answering systems participating in the ResPubliQA 2010 competition, designed to answer questions on a juridical corpora in Romanian, English and French monolingual tasks. Our systems adhere to the classical architecture of a Question Answering system, with an emphasis on simplicity and real time answers: only shallow parsing was used for question processing, the indexes for the retrieval module were built at coarse-grained paragraph level, and the answer extraction component used simple pattern-based rules and lexical similarity metrics for candidate answer ranking.

Keywords: Question Answering, Information Retrieval, ResPubliQA, Legal Documents

1 Introduction

Question answering (QA) is the task of automatically answering a question posed in natural language using a collection of natural language documents. The Natural Language Processing Group of the Faculty of Computer Science, University "Al. I. Cuza" of Iasi, Romania, participated in the Question Answering competitions (QA@Clef) since 2006.

As in 2009, the 2010 QA@CLEF track was called ResPubliQA[2] [9].The structure and the aims of the task remained almost the same as in the previous year: given a pool of 200 independent questions in natural language, participating systems must return an answer for each question. The difference consists in the fact that, in the 2010 competition, besides the JRC-Acquis, a new collection of data was added (the EUROPARL collection) and a new type of questions (the OPINION question type) was defined[3].

[1] University "Al. I. Cuza" of Iasi, Romania
[2] ResPubliQA: http://celct.isti.cnr.it/ResPubliQA/
[3] See RespubliQA task descriptions at http://celct.isti.cnr.it/ResPubliQA/index.php

Preparing the 2010 competition, our main goal was to improve the system built for the 2009 QA@CLEF edition [5], focusing on reducing the running time, while increasing the performances. The best system for the Romanian language participating in RespubliQA 2009 challenge [6] used a sophisticated similarity based model for paragraph ranking, classification and regeneration of the question, considering the EUROVOC terms associated to each document. For English, the best runs produced paragraph rankings considering matching n-grams between question and paragraphs [3]. This retrieval approach seems to be promising, since combined with paragraph validation filters it achieved the best score [10] for English.

One of our main concerns was the improvement of the answer extraction module. Although there are some deep approaches of answer extraction performing well on monolingual QA [4, 11], they are quite demanding in terms of linguistic resources and computational complexity. Since we intended to develop a question answering system for Romanian, easily adaptable to English and French, we adopted a shallow answer extraction method, requiring limited knowledge resources or tools for the three languages. Template-based methods have been used for answer extraction modules, form surface patterns [12], used to match questions with correct answers, to the similarity-based methods that compute the likelihood between a passage and the question, by counting the ratio of question terms occurring in the answer passage [8, 13] or by adopting the IR score of the answer passage as a measure of similarity [7]. Our shallow method for answer extraction, combining the last two approaches, is presented in the next section.

The general architecture of our Question Answering, similar for the three considered languages, is described in Section 2. Section 3 is concerned with the presentation of the results, while the last Section discusses the conclusions and further work envisaged.

2 System components

Similarly to last year's system, we eliminated many pre-processing modules in order to obtain a real-time system. The shallow parsing approaches were considered due to our intention to have a system that does not compromise with the response time, transforming the QA system for online usage (similar to the ENLIGHT system [1] or the SHAPAQA system [2]). After the indexing of the two corpora used as answer extraction collection, which takes about 5 minutes, the time needed for our system to process all the 200 test questions is about 5 seconds: less than 3 seconds for questions pre-processing, about 1 second for snippets extraction, and another second for answer extraction. The main differences from our last year's participation are detailed in the following sections.

2.1 Corpus Pre-processing

The JRC-Acquis corpus is a collection of juridical documents in XML format, with each paragraph numbered. The Europarl[4] corpus represents a collection of the Proceedings of the European Parliament, also in XML format. A subset of the Europarl, containing parallel documents in all the 9 languages involved in the ResPubliQA track (Bulgarian, Dutch, English, French, German, Italian, Portuguese, Romanian and Spanish), was created by crawling the web to get the data from the European Parliament's website. The official nature of the documents forced them to be in a very well organized structure, thus no corpus cleaning was necessary. The only pre-processing performed over the document corpus was its indexing to facilitate the information retrieval module.

2.2 Question Analysis

In this step we identify the semantic type of the answer (expected answer type). A specialized module identifies the question focus, the question type and a set of relevant keywords. The question analyzer performs the following steps:

i. NP-chunking and Named Entity extraction;
ii. Question focus identification (where the focus is the most important word in the question, the clue for determining the answer type);
iii. Question type inferring;
iv. Answer type identification;
v. Identification of the keywords of the sentence. Together with the NPs and named entities, the keywords are to be used by the query generator.

Using GATE Gazetteer, the first step is to identify the named entities in the question. Secondly, the focus of the question is selected from the questions NPs, based on a simple heuristic (the first noun after the wh-word or after the first verb in the sentence, which comes first). For the question analysis, the module developed for the 2009 competition distinguished between *factoid*, *definition*, *purpose*, *reason* and *procedure* question types. In order to address the new requirements regarding the question type identification (the introduction of the *opinion* question type and the merge of the *purpose* and *reason* types), we developed patterns for the new types through empirical analysis of the training question set. Our system currently uses a set of 40 patterns, mainly focusing on the introducing wh-word, in order to classify the questions into their type.

In case of factoid questions, for answer type identification we have 8 answer types: *person*, *number*, *measure*, *location*, *time*, *organization*, *animal* and *object*. For each type, we have created 4-5 derived rules, containing a base and several variables. For instance, for the *location* answer type, we have the following regular expression:

```
qr/(?>regiun.|sectoare|judeţ|staţiun.|loc.|raio.n.|localit..|oraş|capita
l.|insul.|vârf|ţ..?r..?|teritor.|provinci.|continent|locuri|monarhii)/
```

The variable part of the rule was build using WordNet hypernyms of the answer type. Due to its generality, the *object* type is only used if no other rule matched.

[4] Europarl: http://www.europarl.europa.eu/

All the modules used in the question analysis step are pattern-based. Thus, in order to obtain the corresponding question analysis modules for English and French, we adapted the patterns from Romanian, using Google Translate[5] for lexical translations.

2.3 Index Creation and Information Retrieval

The purpose of this module is to retrieve the relevant snippets of text for every question. For this task, similar to our approach for 2009, we used the Lucene[6] indexing and search tools.

i) Query creation

Queries are formed based on the question analysis and it was very similar to the solution we offered last year. They consist mainly of the sequences of keywords previously identified, which are modified using some of the Lucene operators, such as score boosting (the \^" operator, followed by a positive integer), fuzzy matching (the _" operator, followed by a number greater than 0 but less than 1) and the \or" operator (symbolized by words between parentheses).

ii) Index creation

Because of the addition of the EuroParl document collection for this year's competition, we had do devise a way of indexing the corpus in such a way as to include both types of documents. Upon analyzing the corpus, we have determined that a number of properties are common across corpora (all documents have a name, a paragraph number and some text), which allows us to create a single index. We have indexed the corpora at the paragraph level, as our tests have shown that it performs better in terms of relevant text than a document index. Each paragraph in the index is characterized by a filed containing the information described above, and, according to the corpus from which the paragraph was extracted, some extra fields: for the Aquis paragraphs, the extra field is the document name, and for the EuroParl paragraphs, the date of the document's emission and the name of the speaker, in case it is specified. These extra fields are necessary for extracting specific information regarding the answer extraction (for example, in the case of OPINION type questions, the name of the person giving their opinion can be found in the speaker field).

iii) Relevant snippet extraction

Querying over the created indexes, we used the Lucene search engine to extract a ranked list of snippets for every question as possible answer candidates.

2.4 Answer Extraction

In the 2010 year's track, we started from the answer extraction module built in 2009 [5]. We combined in the same class the *reason-purpose* answers and considered a new component for *opinion* answers. For this new component, the candidate answer

[5] Google Translate: http://translate.google.com/

[6] Lucene: http://lucene.apache.org/

ranking prefers answers from the EUROPARL corpus, because it is most likely to contain personal opinions of different political persons. Intuitively, the closeness of two terms may indicate a relation; therefore, we used features based on the distance between the answer and the question terms (keywords, focus, named entities) to obtain a better similarity measurement. The assumption is that, if the candidate answer is close to several keywords or question terms, it is more likely to be relevant. In order to consider the whole set of candidate paragraphs, this similarity score was weighted with the scores provided by the retrieval module for each paragraph.

The most important addition to this module was related to NOA (no answer) cases. Using the training data, we tried to identify the optimum threshold for NOA, the idea being that, if all extracted paragraphs by Lucene have the attached score under this optimal threshold, we consider the final answer to be NOA and the question as not being answered by the system. Otherwise, we apply the same heuristics as last year in order to extract the best answer [5].

For determining the optimal threshold, we considered two values (reflected in the two runs):

- **A higher one** - in this case, the system offers many NOA answers. This value (in our case, for Romanian it was 0.72), was selected so as to lose only a few correct answers, but to have very many questions with NOA answers. This way, the combination between the correct answers and NOA answers offers the highest score in terms of c1 measure.

- **A lower one** – in this case we offer only a few NOA answers. This value (in our case, for Romanian it was 0.67), was selected so we keep the maximum number of correct answers, although giving penalties for the wrong ones.

The thresholds for English and French were very close to the Romanian threshold (for English a little bit higher and for French a little bit lower), differing due to the resources used for these languages in the language extraction module.

3 Results

For the 2010 ResPubliQA track, our team submitted runs for three language pairs: English-English, Romanian-Romanian and French - French. The best runs correspond to the higher threshold we considered for NOA answers, as shown in Table 1.

Table 1: Results of UAIC's runs

	RO-RO		**EN-EN**		**FR-FR**	
answered right	95	102	85	78	54	47
answered wrong	74	93	98	99	124	153
total answered	169	195	183	177	178	200
unanswered right	0	0	0	0	0	0
unanswered wrong	0	0	0	0	0	0
unanswered empty	31	5	17	23	22	0

total unanswered	31	5	17	23	22	0
c@1 measure	**0.55**	**0.42**	**0.46**	**0.43**	**0.30**	**0.24**

Our supposition related to both thresholds was right for Romanian and French, but not for English, where we lost too many correctly identified answers. For example, for Romanian we have in Table 1 on the left column the values obtained for the higher threshold and on the right column values obtained if using the lower threshold. How we can see for higher threshold we lose only a few questions with answers correctly identified by the system (102 – 95 = 7), but the number of NOA is much higher in comparison to its value from right column (31 – 5 = 26). Therefore, the c1 measure is much higher on the left column in comparison with the c1 measure from right column.

4 Conclusions

This paper presents the Romanian Question Answering system which took part in the QA@CLEF 2010 competition. The evaluation shows an overall accuracy of 55% on RO-RO (which is our best results from 2006 till now for Romanian), 46% on EN-EN (which is under result obtained in 2009) and 30 % on FR-FR (which is our first try on this language).

One major improvement made this year was the thresholds used in order to offer NOA answers if the system's confidence is low. Our suppositions related to two classes of thresholds proved correct for Romanian and French, but incorrect for English. Another improvement considered the extraction module, where *reason* and *purpose* question types were combined into one, and a new question class, *opinion*, was introduced.

This year we approached a new language, French, and the main difficulty was the lack of free resources for it. The patterns developed for Romanian were adapted for English and French, as they proved to be of great importance.

Acknowledgements. The research presented in this paper was funded by the Sector Operational Program for Human Resources Development through the project "Development of the innovation capacity and increasing of the research impact through post-doctoral programs" POSDRU/89/1.5/S/49944.

References

1. Akerkar, R., Joshi, M.: Natural Language Interface Using Shallow Parsing. International Journal of Computer Science & Applications, Vol. 5, Issue 3, pp 70-90, Canada (2008)
2. Buchholz, S., Daelemans, W.: SHAPAQA: Shallow Parsing for Question Answering on the World Wide Web. In Proceedings Euroconference Recent Advances in Natural Language Processing. Pp. 47-51, 5-7 September, Tsigov Chark, Bulgaria (2001)
3. Correa, S., Buscaldi, D. and Rosso, P. NLEL-MAAT at CLEF-ResPubliQA. Working Notes for the CLEF 2009 Workshop, 30 September-2 October, Corfu, Greece (2009)

4. Cui, H., R. Sun, K. Li, M.Y. Kan, and T.S. Chua, "Question answering passage retrieval using dependency relations," in Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, 2005, pp. 400-407.
5. Iftene, A., Trandabăţ, D., Pistol, I., Moruz, A. M., Husarciuc, M., Sterpu, M. and Turliuc, C.: Question Answering on English and Romanian Languages. In Proceedings of the CLEF 2009 Workshop. 30 September - 2 October. Corfu, Greece. (2009)
6. Ion, R., Ştefănescu, D., Ceauşu, A., Tufiş, D., Irimia, E. and Barbu-Mititelu, V. A. Trainable Multi-factored QA System. Working Notes for the CLEF 2009 Workshop, 30 September-2 October, Corfu, Greece (2009)
7. Kwok, K.-L. and P. Deng, P., "Chinese Question-Answering:Comparing Monolingual with English-Chinese Cross-Lingual Results," in Asia Information Retrieval Symposium, 2006, pp. 244-257.
8. Molla, D. and M. Gardiner, M., "AnswerFinder — Question Answering by Combining Lexical, Syntactic and Semantic Information," in Australasian Language Technology Workshop (ALTW) 2004, Sydney, Australia, pp. 9-16.
9. Peñas, A., Forner, P., Sutcliffe, R., Rodrigo, Á., Forăscu, C., Alegria, I., Giampiccolo, D., Moreau, N., and Osenova, P.: Overview of ResPubliQA 2009. Question Answering Evaluation over European Legislation (2009)
10. Rodrigo, Á., Pérez, J., Peñas, A., Garrido, G. and Araujo, L. Approaching Question Answering by means of Paragraph Validation. Working Notes for the CLEF 2009 Workshop, 30 September-2 October, Corfu, Greece (2009)
11. Shen, D., G. Saarbruecken, and D. Klakow, "Exploring Correlation of Dependency Relation Paths for Answer Extraction," in Proceedings of ACL 2006, 2006, Sydney, Australia, pp. 889-896.
12. Soubbotin, M.M. and S.M. Soubbotin, "Patterns of Potential Answer Expressions as Clues to the Right Answers," in Proceedings of the Tenth Text REtrieval Conference (TREC 2001), 2001, Gaithersburg, MD, pp. 134-143.
13. Zhao, Y., Z.M. Xu, Y. Guan, and P. Li, "Insun05QA on QA track of TREC2005," in TREC, 2005, Gaithersburg, MD.