Question Answering for Machine Reading Evaluation on Romanian and English

Adrian Iftene[1], Alexandru-Lucian Gînscă[1], Alex Moruz[1, 2], Diana Trandabăţ[1, 2], Maria Husarciuc[3]

[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, lucian.ginsca, amoruz, dtrandabat, mhusarciuc}@info.uaic.ro

Abstract. This paper describes UAIC[1]'s Question Answering for Machine Reading Evaluation systems participating in the QA4MRE 2011 evaluation task. The system is designed to extract knowledge from large volumes of text and to use this knowledge to answer questions in Romanian and English monolingual tasks. Our systems were built on the architecture of a Question Answering system, customized for this new task. Thus, the new system used from our previous question answering systems the question processing and information retrieval components, adapted for new requests. Additionally, a new component was added in order to detect the most probable answer of a question, from a list of possible answers.

Keywords: Question Answering for Machine Reading Evaluation, Information Retrieval

1 Introduction

Question Answering for Machine Reading Evaluation (QA4MRE[2]) is the exercise of developing a methodology for evaluating Machine Reading systems through Question Answering and Reading Comprehension Tests.

In 2011, the QA4MRE task focused on reading a single document and correctly identifying the answer from a set of possible answers, using some inference and the previously acquired background knowledge. The competitors received test data and background knowledge related to three topics: *AIDS*, *Climate Change* and *Music and Society*. An important note is that, for all involved languages (English, Spanish, German, Italian and Romanian), the test data was the same (parallel translations) and the background knowledge was available to all participants.

Preparing the 2011 exercise, we started from the systems built for the 2009 and 2010 QA@CLEF editions [1], [2].

---

[1] University "Al. I. Cuza" of Iasi, Romania
[2] QA4MRE: http://celct.fbk.eu/QA4MRE/index.php

The general architecture of our Question Answering for Machine Reading Evaluation system, similar for the two considered languages, is described in Section 2. Section 3 is concerned with the presentation of the results, while the last Section discusses the conclusions.

2 System components

The system we participated with in QA4MRE 2011 uses some components from the system we used in 2010 [2], adapted for the new task (*question analysis*, *corpus indexing* and *snippet extraction*) and some new components (mainly for the *identification of the correct answer*). The architecture of the current Romanian system is presented in Figure 1 and the main components are detailed in next subsections.

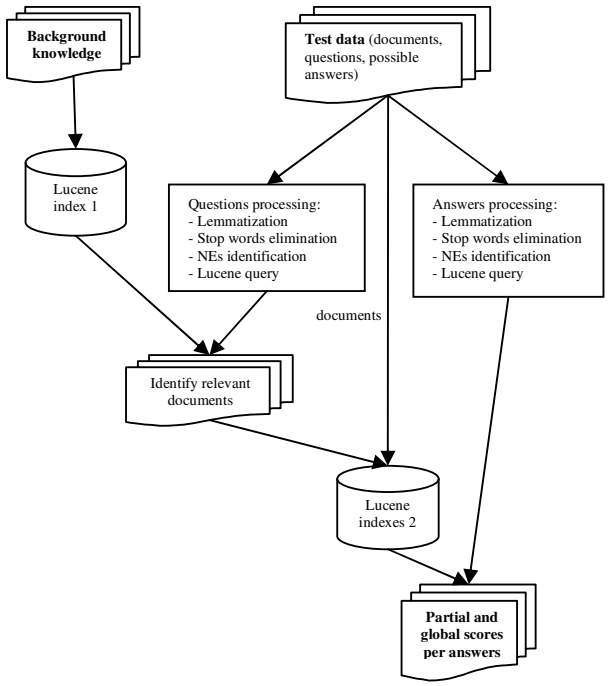

Figure 1: UAIC system used in QA4MRE

The English system is almost similar with the Romanian system. Due to technical problems in handling the English background knowledge, we skipped over using the components presented in subsections 2.1 and 2.3, and the component detailed in subsection 2.4 was only partially used.

2.1 Background knowledge indexing

The Romanian background knowledge consists of a collection of 161,279 documents in text format (25,033 correspond to the AIDS topic, 51,130 to Climate Change topic and 85,116 to Music and Society topic). The indexing component was responsible for taking the name of the file and the text from it and adding both to the *Lucene index 1* using Lucene[3] libraries [3] (see Figure 1 for details).

2.2 Test data processing

The test data consists in an XML file with 12 test documents (4 documents for each of the three topics), 10 questions for each document (120 questions in total) and 5 possible answers for each question (600 possible answers in total).

Test data processing involved 3 operations: (a) extracting documents, (b) processing questions and (c) processing possible answers. For (a), we extract the content of the tag <*doc*> from the XML with the test data, and save it in a relative path corresponding to the current <*topic id*> and <*reading test id*> tags. In this way, we can use later this document to build *Lucene index 2*, as presented in Figure 1.

For (b) and (c), we used our question processing module from [1] and performed the following important steps:

i. Stop words elimination;
ii. Lemmatization;
iii. Named Entity identification;
iv. Lucene query building.

For the first three steps, we used this year the web services available both for Romanian and English from the Sentimatrix[4] project [4].

For instance, in the case of the question "*Ce a spus Nelson Mandela la conferința de presă*?" (En: *What did Nelson Mandela say at the press conference*?) the execution of the above steps has the following results:

- in the first step, the following stop words are eliminated: *ce, a, la, de* (En: *what*, *the*, *at*);
- in the next step, lemmas for the words *spus*, *conferința, presă* (En: *say, conference, press*) are identified;
- in the third step, *Nelson Mandela* is identified as a Named Entity;
- in the last step, the Lucene query is build: *"(spus^2 spune) Nelson^3 Mandela^3 (conferința^2 conferință) (presă^2 presa)*".

From the above Lucene query, one can notice that we consider named entities to be of most relevance (hence receiving a boot of 3), and the inflected and lemmatized form of the words existing in the question receive a lower boost value (2 in the example above).

[3] Lucene: http://lucene.apache.org/
[4] Sentimatrix: http://www.sentimatrix.eu/

Additionally for (c), we used from the ontology presented in [5] the relations between regions and cities and the relations between cities and countries, in order to eliminate the answers with low probability to be the final required answer. Thus, for the question *În ce orașe europene a cântat Annie Lennox*? (En: *In which European cities has Annie Lennox performed*?), we eliminate from the list of possible answers the answers with non-European cities.

2.3 Information Retrieval on Background Knowledge

The purpose of this module is to retrieve, for every question, the relevant documents from the background knowledge. For this task, similar to our previous approach from 2010 and 2009, we used the Lucene search over the index presented in section 2.1 using the Lucene queries presented in section 2.2.

The result of this step is a list of documents from the background knowledge, with associated relevance score obtained after performing Lucene search using the queries obtained after processing the questions. Thus, we have *Score*(*d, q*), the relevance score for a document *d* when we search the background knowledge with the Lucene query associated to question *q*.

Similar to 2.2 (a), we copy the content of all the files from this list in a relative path with the name obtained from the <*topic id*>, <*reading test id*> and the <*question id*> tags.

2.4 Indexing and searching using relevant documents for questions

This module takes all documents from 2.2 (a) and 2.3 (using the relative path obtained from <*topic id*>, <*reading test id*> and <*question id*> tags) and puts them in a separate index. The results of this step are 120 separate indexes for every question from the initial test data (*Lucene index 2* in Figure 1). Because of how we saved the relevant files using relative paths, files from the Lucene index 2 are relevant to the corresponding question for the specific relative path.

Then in every index, we performed searches using Lucene queries obtained at 2.2 (c) and, for every answer, a list of documents with Lucene relevance scores are returned, where *Score*(*d, a*) is the relevance score for document *d* when we search with the Lucene query associated to the answer *a*.

2.5 Identifying of most probable answer

The results of this step are the runs submitted by our group. In order to do this for every question, we combine the Lucene scores from 2.3 and 2.4 using the following formula:

$$Score(a) = \sum_{d \in Relevant\_docs\ for\ q} Score(d, q) \times Score(d, a)$$

where *a* is the current answer, *q* is the corresponding question from the test data and *d* is a document from the list of relevant documents returned by the search in *Lucene index 2*.

After we calculate the above value for all answers associated to a question, we consider the answer with the highest value as being the most probable answer.

3 Results and Evaluation

For the QA4MRE 2011 task, our team submitted 10 runs, out of which 9 were for the Romanian-Romanian language pair and one for the English-English pair.

The evaluation of this year's results is done from two different perspectives. The first one is equivalent to a traditional evaluation in which all the answers are gathered in a single set which is then compared to a gold standard, not taking into account the document associated with a particular answer. On the other hand, the reading perspective offers insight on how well the system "understands" a particular document. At first, the C@1 measures [6] of each test comprising of 10 questions per document are taken into consideration. These results are then used to obtain statistical measures, such as the mean, median and standard deviation over values grouped by topic or as an overall view.

3.1 Evaluation at the question answering level

We grouped our runs for Romanian based on the threshold used to consider a NOA response. For the first group of runs, we imposed the condition that NOA should be used for the case in which the Lucene index searcher didn't return any documents. For the second and third group of runs, the threshold was determined by the value given by the Lucene score associated with each document found. For the second group, the threshold was set at 0.05 and for the third one, at 0.02. The best results obtained by runs from each of the three groups are presented in Table 1 and Table 2 in the Ro-Ro column, which has three sub columns. For the English run, the threshold was fixed at 0.02.

Table 1: Results of UAIC's runs at question answering level

	Ro-Ro			**En-En**
answered right	30	11	19	25
answered wrong	85	19	43	47
total answered	115	30	62	72
unanswered right	0	19	11	12
unanswered wrong	0	66	42	34
unanswered empty	5	5	5	2
total unanswered	5	90	58	48
Overall accuracy	**0.25**	**0.09**	**0.16**	**0.21**
C@1 measure	**0.26**	**0.16**	**0.23**	**0.29**

As can be seen in Table 1, the best result of our system in terms of C@1 measure is obtained for the English run. For Romanian, the best run is the one in the first group, and the worst results are from the group with the 0.05 threshold.

We can observe the influence of the correctly unanswered questions in the C@1 measure when comparing the number of right answers for the best run for Romanian, the one in the first column, with the one for the English run. Although in the Ro-Ro run, a higher number of questions were correctly answered (30 right answers) than in the En-En run (25 right answers), the better C@1 measure is obtained for the English run. This is explained by the difference in the number of correctly unanswered questions. Thus, for the Romanian version of the system, we have decided to only return an empty answer when our information retrieval module did not return any result (this was empirically determined); this greatly improved the precision of determining empty answers, but significantly reduced our recall for them.

In the case of the English system, we have empirically established that the Lucene threshold below which we can safely assume that no answer can be provided is the score 0.02. As can be seen from the table given above, this resulted in many more unanswered questions, which greatly decreased precision but improved the recall, and also resulted in a significant increase in the C@1 measure.

3.2 Evaluation at the reading test level

In Table 2, we present the median and mean for each of the three topics, Topic1 (AIDS), Topic2 (Climate Change) and Topic3 (Music and society) and their overall values. The three columns in the Romanian part of the table correspond to the best means given by runs in each of the three groups described in the previous section.

Table 2: Results of UAIC's runs at reading test level

	RO-RO			**EN-EN**
Topic1 median	0.10	0.00	0.07	0.23
Topic2 median	0.40	0.00	0.29	0.31
Topic3 median	0.30	0.32	0.33	0.36
Overall median	**0.20**	**0.00**	**0.16**	**0.31**
Topic1 mean	0.10	0.04	0.08	0.25
Topic2 mean	0.39	0.08	0.26	0.27
Topic3 mean	0.29	0.30	0.31	0.32
Overall mean	**0.26**	**0.14**	**0.22**	**0.28**

These results are consistent with the trend introduced in Table 1. The best mean was obtained for the English run, followed by the best Romanian run from the first threshold group of runs.

One anomaly can be observed in the results of the English run. In all the Romanian runs, the median is significantly lower than the mean, but in the English run, the order is reversed. We can therefore consider that our system performs uniformly well on the majority of the test for the English run, with fewer spikes in the C@1 values distribution.

3.3 Error analysis

In extension to the analysis carried out above, we have also performed error analysis over the reported results (only the top scoring runs were analyzed). One of the most common error sources arises from our attempt to take into account all of the supporting snippets that our information retrieval procedure returns. Instead of comparing the results for all the snippets independently, we have combined the results for each candidate answer by calculating the average score, which is to say, if a candidate answer had more than one supporting snippet, the score for the entire candidate was the average of all the scores of its supporting snippets. Examples of this type of error can be seen for questions 8, 9, 2 and 3 in *Topic 1*, *Reading Test 1*. The solution to this type of error is to only take into account the highest scoring snippet for each candidate, instead of combining the scores.

Another error source we have indentified is incorrect query building, especially in the case of long queries, as can be seen in the case of question 5 in *Topic 1*, *Reading Test 1*. The mistake in query building arises from the fact that for the second candidate, the preposition and article construct "*într-un*" (En: *in a*) is not recognized as a stop word, and as such is included in the query, thus artificially boosting the query score (the Lucene query we have used is "*asistentă (medicală^2 medical) într-un^2 spital*"). The solution to this error is to perform more accurate POS tagging and to exclude functional words from queries.

Ambiguity in terms of answer extraction is also a cause for errors, as can be seen in question 6 in *Topic 1*, *Reading Test 1*. After the information retrieval step, two candidates (the first and the third) are determined to have identical top scores, supported by identically scoring snippets. In this case the QA system defaults to choosing the first candidate, which is incorrect. The solution to this type of error is twofold: performing information extraction at the paragraph or sentence level, in order to only take into account those parts of the knowledge base which refer to the question focus, and to perform an additional step of determining the distance between each candidate and the focus of the question in the knowledge base.

The same solution as the one described above can be applied to error cases such as those found for question 4 in *Topic 2*, *Reading Test 8*. This error comes from the fact that the answer candidates are all country names, and the top scoring snippet is obtained for the name that has the highest Tf/Idf value, regardless of the relevance to the original question. This type of error is quite common, and is not limited to single entity candidates, as can be seen in the case of questions 7, 9 and 10 in *Topic 2, Reading Test 8* and for question 7, *Topic 3, Reading Test 12* (for the En-En task). Another possible solution for this issue is to increase the score of those candidates which can be found in multiple supporting documents in the knowledge base.

In some cases, query generation requires the use of semantic equivalents for candidates in order to determine the correct answer, as is the case for question 10 in *Topic 1, Reading Test 1*. The system chooses an incorrect answer because the correct candidate, in this case "*categoric da*" (En: *definitely yes*) cannot be found in the supporting document or in the relevant articles of background knowledge in the same surface form, but it can be found as a semantic equivalent. In order to compensate for this problem, query generation should also take into account semantic equivalents of words.

Some error cases are due to the fact that, in some cases, the answer extraction module does not choose the top scoring snippet, and therefore misses the correct answer by discarding it, as can be seen for question 2 in *Topic 3, Reading Test 12* (for the En-En task).

An error that is only encountered in the case of the En-En task is that of missing background information. Because of time constraints, we were unable to make use of the BK available for the English task, and, because of this, we were unable to find some answers, as can be seen in question 3, *Topic 3, Reading Test 12*, where the correct answer, "*five*", is not present at all in the supporting document. This can also be seen in the case of question 8, *Topic 3, Reading Test 12*, where the reference to a person born in 1889 is missed because the supporting document makes no reference to that year.

Numbers are also a major cause of errors, mainly because they can be written either with letters or with digits, as can be seen in question 4, *Topic 3, Reading Test 12*. This can be solved at the question processing stage, where the numbers can be transformed in both formats, in order to cover all possibilities.

Some errors also come from the fact that the Lucene indexer and query system treats the query words and the indexed text as a bag of words, and disregards the fact that, in some cases, if the query words are in different sentences, they lose their meaning as an answer candidate. This is the case for question 9, *Topic 3, Reading Test 12*, where instead of searching for "35 years" or "50 years", the system instead searches for "35" "years" and "50" "years". Since "50" appears in the text twice, and "35" only once, the selected candidate is "50 years" (on the basis of higher Tf/Idf), regardless of the act that "50" is in no way connected to "years".

4 Conclusions

This paper presents our systems built for the Question Answering for Machine Reading Evaluation task within CLEF 2011 labs. The evaluation shows an overall accuracy of 0.25 for the Ro-Ro monolingual task and 0.21 for the En-En task, and a C@1 measure of 0.26 for Ro-Ro and 0.29 for En-En. The thresholds used to obtain the NOA answers were properly selected for English (from this reason the C@1 measure is higher for English, even if the overall accuracy is higher for Romanian).

The presented systems were built starting from the main components of our QA systems (the question processing and information retrieval modules) to which new components were added for identifying the most probable answer from a set of possible answers.

What is interesting is that, although we did not use for the English monolingual task the background knowledge, the results were better in terms of the C@1 measure for English, because of the threshold we considered for the NOA answers.

Acknowledgement. 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. Iftene, A., Trandabăţ, D., Moruz, A., Pistol, I., Husarciuc, M., Cristea, D.: Question Answering on English and Romanian Languages. In C. Peters et al. (Eds.): CLEF 2009, LNCS 6241, Part I (Multilingual Information Access Evaluation Vol. I Text Retrieval Experiments). Pp. 229-236. ISBN 978-3-642-15753-0. Springer, Heidelberg. (2010)
2. Iftene, A., Trandabăţ, D., Moruz, A., Husarciuc, M.: Question Answering on Romanian, English and French Languages. Notebook Paper for the CLEF 2010 LABs Workshop, ISBN 978-88-904810-0-0, ISSN 2038-496322-23, 22-23 September, Padua, Italy. (2010)
3. LUCENE: http://lucene.apache.org/java/docs/.
4. Gînscă, A. L., Boroş, E., Iftene, A., Trandabăţ, D., Toader, M., Corîci, M., Perez, C. A., Cristea, D.: Sentimatrix - Multilingual Sentiment Analysis Service. In Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (ACL-WASSA2011). ISBN-13 9781937284060, Portland, Oregon, USA, June 19-24. (2011)
5. Iftene, A., Balahur-Dobrescu, A.: Named Entity Relation Mining Using Wikipedia. In Proceedings of the Sixth International Language Resources and Evaluation (LREC'08). ISBN: 2-9517408-4-0, EAN: 9782951740846. 28-30 May, Marrakech, Morocco. (2008)
6. Peñas, A., Rodrigo, A.: A Simple Measure to Assess Non-response. In Proceedings of 49th Annual Meeting of the Association for Computational Linguistics - Human Language Technologies (ACL-HLT 2011), Portland, Oregon, USA, June 19-24. (2011)