University of Wolverhampton at CLEF 2007*

Georgiana Puşcaşu and Constantin Orăsan

Research Group in Computational Linguistics
University of Wolverhampton, UK
{georgie, C.Orasan}@wlv.ac.uk

Abstract. This paper reports on the participation of the University of Wolverhampton in the Multiple Language Question Answering (QA@CLEF) track of the CLEF 2007 campaign. We approached the Romanian to English cross-lingual task with a Question Answering (QA) system that processes a question in the source language (i.e. Romanian), translates the identified keywords into the target language (i.e. English), and finally searches for answers in the English document collection. We submitted one run of our system that has achieved an overall accuracy of 14%, and a precision over non-NIL answers of 33.73%. Error analysis revealed that this low performance is mainly due to the lack of a reliable translation methodology from the source in the target language.

1 Introduction

Cross-lingual Question Answering is defined as the task of retrieving the answer in one language (the target language) to a question posed in a different language (the source language). Last year, a new Romanian-to-English (RO-EN) cross-lingual QA task was organised for the first time within the context of the CLEF campaign [10], and it consisted of retrieving answers to Romanian questions from a collection of English documents. This year's task [6] was similarly organised, with the exception that all questions were clustered in classes related to the same topic, some of which even contain anaphoric references to other questions from the same topic class, or to their answers. Besides the usual news collections employed in the search for answers, this year's novelty was the fact that Wikipedia articles could also be used as answer source.

This is the first time a Romanian-English cross-lingual QA system fully developed at the University of Wolverhampton has participated in the QA@CLEF competition. This system contains the classical QA modules: question processing, information retrieval and answer extraction [7]. In addition, the cross-lingual capabilities are provided by a Romanian-to-English term translation module. This paper describes the development stages and evaluation results of our system. The rest of the paper is organised as follows: Section 2 provides an overall description of the system, while Sections 3, 4, 5 and 6 present the four embedded modules - the question processor, term translator, passage extractor and answer extractor respectively. Section 7 captures the evaluation results and their analysis. Finally, in Section 8, conclusions are drawn and future directions of system development are considered.

* This work has been supported by the EU funded project QALL-ME (FP6 IST-033860).

2 System Overview

Question Answering systems normally share a pipeline architecture consisting of three main stages: question analysis, passage retrieval and answer extraction [7]. For cross-lingual systems, the language barrier is usually crossed by employing free online translation services for translating the question from the source language into the target language [8, 14]. The QA process is then entirely performed in the target language by a monolingual QA system. A different approach taken by some cross-lingual systems automatically translates the document collection in the source language, performs monolingual QA in the source language [2], and then converts the answer back into the target language by aligning it with the corresponding span in the original document. Another alternative approach involves monolingual QA in the source language and then translating the answer, but this approach is feasible only when document collections covering the same material are available in both the source and target languages [1].

Since we could not identify reliable translation services from Romanian into English for translating complete questions, nor English-Romanian full document translation tools, the first two approaches were discarded. For the third option, the impediment was the lack of a Romanian document collection equivalent to the English one. Therefore we adopted a slightly different methodology where the question analysis is performed in the original source language without any translation in order to overcome the negative effect of full question translation on the overall accuracy of the system. Afterwards, in order to link the two languages involved in the cross-lingual QA setting, term translation is performed by means of bilingual resources and linguistic rules. The search for passages and answers is then performed in the target language documents. This method was also employed by Sutcliffe et al. [13] and Tanev et al. [14].

The system architecture consists of a four-module pipeline, where each module is responsible for a different stage in answering a question. These four modules are:

1) ***Question Processor***
   This module analyses each Romanian question in order to identify the type of the question and of the expected answer, the question focus, and all relevant keywords.
2) ***Term Translator***
   For each question term, all translation equivalents are generated by consulting bilingual resources and by employing linguistic rules to assemble individual words into target language terms.
3) ***Passage Extractor***
   At this stage candidate snippets of text are retrieved from the English document collection on the basis of a query that includes the translation equivalents of all terms identified in the question.
4) ***Answer Extractor***
   On the basis of the information extracted by the Question Processor, this module identifies in the previously retrieved snippets a set of candidate answers matching the expected answer type. One answer is then selected by ranking the resulting set of candidate answers.

The following four sections present in more detail the functionality of each module.

3 Question Processor

This module is mainly concerned with the identification of the semantic type of the entity sought by the question, but it also provides the question type, focus, and relevant keywords. To achieve these goals, our question processor performs the following steps:

a) *Question Annotation*: The questions are first morpho-syntactically pre-processed using the TnT POS tagger [3] trained on Romanian [16], and afterwards noun phrases (NPs) and named entities (NEs) are identified using a rule-based approach. Temporal expressions (TEs) are also detected using the adaptation for Romanian of an English TE identifier and normalizer [11].

b) *Question Focus Identification:* The question focus is considered to be either the noun determined by the question stem or the head noun of the first question NP if this NP comes before the question's main verb or if it follows the verb *"to be"*.

c) *Distinguishing the Expected Answer Type (EAT)*: Our system can detect the following expected answer types: PERSON, LOCATION, ORGANIZATION, TEMPORAL, NUMERIC, DEFINITION and GENERIC. The assignment of a class to an analysed question is performed using the question stem and the question focus type. The latter is obtained using Romanian WordNet [17] sub-hierarchies specific to the categories PERSON / LOCATION / ORGANIZATION.

d) *Inferring the Question Type*: This year, the QA@CLEF main task distinguishes among four question types: *factoid*, *definition*, *list* and *temporally restricted* questions [6]. As temporal restrictions can constrain any question type, we first detect whether the question has the type *factoid*, *definition* or *list*, and then search for temporal restrictions. The question type is identified as follows: for questions which ask for definitions of concepts, the assigned question type is *definition*; if the question focus is a plural noun, then the question type is *list*, otherwise it is *factoid*. The temporal restrictions are identified using several patterns and the information provided by the TE identifier.

e) *Keyword Set Generation*: The set of keywords is generated by listing the question terms in decreasing order of their relevance, as follows: the question focus, the identified NEs and TEs, the remaining NPs, and the non-auxiliary verbs. This relevance ranking is not currently employed at the retrieval stage, but it will be used in the future to assign weights to each term. Given the grouping of questions into topics and the presence of anaphoric expressions between same topic questions, a shallow anaphora resolution mechanism is employed to expand the keyword set with other possibly relevant terms as described below. The expanded set of keywords is then passed on to the term translation module, in order to obtain English keywords for passage retrieval.

f) *Resolution of anaphoric expressions*: As related questions are organised in clusters, in a number of cases, the links between questions are realised using anaphoric pronouns, and therefore, in order to obtain a more complete list of keywords, anaphora resolution is necessary. Given the difficulty of the task, it is not possible to employ a fully fledged anaphora resolution system. Instead, the set of keywords related to a question is expanded with the list of NEs present in the cluster. This is done for two reasons. On the one hand, investigation of the question clusters revealed that pronouns often refer to NEs in the cluster. On the other hand, given that the questions are related, it is possible

that NEs present in the questions also co-occur in the same document. As a result, it is more likely to extract relevant documents with this expanded query. Certain questions referred to the answer of the previous question. Currently, this problem is not addressed because in our present system there is no way to feed an answer back into the system.

4 Term Translator

Each keyword is translated into several translation equivalents, which are then grouped using the disjunction operator into a keyword specific sub-query. The conjunction of all sub-queries corresponding to the question keywords forms the final query.

Term translation is achieved with an approach similar to the one we employed when we participated together with two Romanian research groups in the same task at CLEF 2006 [12]. It also resembles the one employed by Ferrandez et al. [5] for the English to Spanish task of the same CLEF campaign. This method employs WordNet and the ILI alignment between the English WordNet and the other WordNets developed in the EuroWordNet and BalkaNet projects. The underlying idea is that, given a Romanian word, the Romanian WordNet and its alignment to the English one, we identify all possible translations of the word by finding all the synsets it appears in and extracting the equivalent English synsets through the ILI alignment. If the word to be translated does not appear in the Romanian WordNet, as is quite frequently the case, we search for it in other dictionaries and preserve the first three translations. If still no translation is found, the word itself is considered as translation, an approach which works reasonably well for NEs. In the case of multi-word terms, each word is translated individually using the method described above. After that, rules are employed to convert the Romanian syntax into English syntax, and to obtain the translation equivalents of a given term.

One drawback of this method is that, by not employing word sense disambiguation, it proposes too many translations for a word. To address this problem, we implemented a ranking method which relies on parallel English-Romanian Wikipedia pages and on the assumption that the two sets of pages will contain more or less the same information, so it will be possible to find the most likely translation for a given term. Unfortunately, preliminary experiments revealed that, by including this approach, a very small number of passages are retrieved, many of which do not contain the answer to the question. Due to time restrictions, we were unable to properly tune the method to retrieve better passages, and for this reason we did not employ it in this year's submission.

5 Passage Extractor

The purpose of this module is to extract a list of passages which may contain the answer to a given question from the following three document collections: English Wikipedia pages collected in November 2006, Los Angeles Times from 1994 and Glasgow Herald from 1995. This is the first time that Wikipedia has been included in the document collection and, as a result of the fact that it is several orders of magnitude bigger than the other two collections, the search space was significantly larger than in previous years, making the task more difficult. Given that the documents in each collection are formatted in different ways, each had to be indexed individually and processed in a

slightly different manner. For indexing and retrieval, we used Lucene [9], an open source information retrieval library.

Passages are extracted using the query proposed by the term translation module, including all possible translations of the question keywords. In the initial experiments we limited the number of translations used for each original keyword, but as a result, the number of retrieved snippets was too low. This can be explained by the fact that no disambiguation was performed and therefore it was possible that some of the translations were ranked high and included in the query, even though they were not appropriate. As the attempt to order the translations according to their likeliness of being the correct translation of a keyword did not lead to satisfactory results, it is not used in this year's submission. In light of this, we decided to consider all the translations identified for a keyword and link them with the `OR` operator provided by Lucene.

We indexed the collection in order to retrieve documents containing the keywords, and not actual passages. This approach is taken because it offers more flexibility and allows better control of the methods which retrieve candidate passages. It has the drawback that it needs to process each document individually and extract relevant passages. For this year's system only sentences are extracted. In order to do this, each sentence from the retrieved documents is scored on the basis of how many keywords, TEs and NEs it contains. At present, up to 25 sentences with the highest scores are retrieved from each document, provided that their score is higher than a predefined threshold. This set of sentences is fed into the next module, the answer extractor.

6 Answer Extractor

Once candidate answer-bearing document passages have been selected, the answer extractor starts by merging all passages retrieved for questions belonging to a certain topic. All retrieved passages are parsed with Conexor's FDG Parser [15] and with the NE identifier embedded in the GATE toolkit [4]. A question-based passage ranking is then applied to the merged set of passages to identify the most relevant passages. The answer extractor then addresses each EAT in a different manner, as follows:

a) *Expected answer type is a Named Entity*: Named entities having the desired answer type are identified in the retrieved passages and added to the set of candidate answers. Candidate answers are then ranked on the basis of the passage score, the distance to other keywords and their frequency. The candidate answer with the highest score is presented as final answer. When the retrieved passages contain no candidate answer, the system returns NIL.

b) *Expected answer type is NUMERIC*: Several NUMERIC answer sub-categories are distinguished: MONEY, PERCENTAGE, MEASURE and NUMERIC-QUANTITY (any other NUMERIC entity). Patterns are defined for exact candidate answer identification, patterns that take into consideration either the format of certain numeric expressions or the presence of the question focus in the neighbourhood of a numeric expression. The process of ranking candidate answers relies on the same parameters as in the case of the Named Entity answer type.

c) *Expected answer type is TEMPORAL (i.e. a Temporal Expression)*: The subtypes of TEMPORAL entities that guide the answer extraction process are: MILLENNIUM, CENTURY, DECADE, YEAR, MONTH, DATE, TIME, DURATION (applying also to questions asking about age) and FREQUENCY. If the granularity of the expected answer is coarser than the granularity of a candidate answer TE, patterns are employed to convert the TE to the required granularity (e.g. if the EAT is YEAR and the candidate answer has the granularity DATE like *"25th of January 1993"*, then only *"1993"* is extracted).

d) *Expected answer type is GENERIC*: When the EAT is neither a NE, nor a NUMERIC or TEMPORAL entity, the question focus is essential in finding the answer. The candidate answers are constrained to be hyponyms of the question focus head.

e) *Expected answer type is DEFINITION*: A different approach is taken when the question asks for the definition of a concept. Wikipedia contains definitions for a large number of concepts, therefore our first attempt is to obtain the definition from the Wikipedia page corresponding to that concept. To this end, Lucene is used to return Wikipedia pages which contain in their title words from the concept to be defined. Because this approach returns more than one document, a ranking method is applied to the retrieved documents. The more concept words the document title contains, the more the document score gets boosted. Once the documents are ranked, patterns are used to locate the answer. Whenever no answer can be located in Wikipedia, passages are extracted from the other two document collections using the passage extractor described in Section 5 and the regular expressions are then applied to them. Unfortunately, this fall-back approach performed quite poorly.

7 Evaluation Results

This section describes the results corresponding to the run we submitted for the RO-EN QA task at CLEF-2007. The methodology employed targets precision at the cost of recall, by providing NIL answers to those questions we cannot reliably locate a candidate answer in the retrieved passages. Apart from this, no more than one answer per question is returned, and this is the first ranked answer, when it can be identified.

Table 1 illustrates the detailed results achieved by our system. Despite the fact that our Question Processor is able to recognise questions asking for LISTs, the answer extractor does not tackle this type of questions. The overall accuracy of our system was evaluated at a generic score over all questions of 14%. An analysis of the system output revealed the fact that our system was unable to locate an answer and returned the answer NIL for 117 questions. It retrieved 83 answers, out of which 28 correct, 49 wrong, 4 unsupported and 2 inexact.

A preliminary analysis of the incorrect and NIL answers showed that their main cause was the poor translation of the question keywords, this yielding either irrelevant or no passages being retrieved from the English document collection. If we consider the fact that our system, whenever it has little or no confidence that it has found a correct answer, does not attempt to answer the question by returning NIL, and we analyse only the answers retrieved by the system, the conclusion is that out of 83 answers, 28 are correct, this yielding a precision of 33.73%.

	FACTOID	LIST	DEFINITION	TEMPORALLY RESTRICTED
RIGHT	15	0	13	0
WRONG	140	9	17	2
UNSUPPORTED	4	0	0	1
INEXACT	2	0	0	0
TOTAL	161	9	30	3
ACCURACY	**9.32%**	**0.00%**	**43.33%**	**0.00%**

Table 1. Detailed evaluation results

Unsupported answers are correct answers, but the returned support passage is not considered relevant enough for the question. Given that we can not access the correct answers and expected support passages, it is difficult to judge whether the four retrieved passages are appropriate or not. For example, in the case of the question *"What kind of animal did Victor Bernal try to buy on the 25th of January 1993?"*, our returned answer was *"gorilla"*, and it was extracted from: *"The sting took place on Jan. 25, 1993, when Bernal and the others were escorted onto a DC-3 cargo plane parked in a remote corner of a small Miami airport to see the gorilla, crated for shipment.*", which seems correct, but probably it can not be justified only by the presence of Bernal's name, of the date mentioned in the question, and of the noun *"gorilla"*, which is a type of *"animal"*.

In the case of inexact answers, the answer-string contains the correct answer and the provided snippet supports it, but the answer-string is incomplete or more detailed than the correct answer. For example, given the question *"What is the occupation of Michael Barrymore?"*, our inexact answer was *"troubled comic"* and the supporting passage was *"Troubled comic Michael Barrymore last night received an ovation as his show, Strike It Lucky, was named Quiz Programme of the Year at the National Television Awards.*". Most of these errors can be corrected by improving the answer extractor with more specific rules as to the extent of the required answer.

8 Conclusions

This paper describes the development stages of our cross-lingual Romanian to English QA system that participated in the QA@CLEF campaign. Adhering to the generic QA architecture, our system implements the three essential stages (question processing, passage retrieval and answer extraction), as well as a term translator which provides cross-lingual capabilities by translating question terms from Romanian into English. This year our emphasis was less on fine tuning the system, and more on exploring the issues posed by the task and developing a complete system able to participate in the competition. Therefore, all four modules are still in a preliminary stage of development.

The run we submitted for the Romanian to English cross-lingual QA task achieved an overall accuracy of 14%, the best score achieved among systems with English as target language [6]. An in-depth analysis of the results at different stages in the QA process has revealed a number of future system improvement directions. The term translation module has a crucial influence over the systems performance, and will therefore receive most of our attention. Apart from this, we will further investigate the

ranking method for translation equivalents which relies on information from parallel English-Romanian Wikipedia pages in order to improve its performance, as we believe it is a promising research direction. We also intend to improve our answer extraction module by identifying a better answer ranking strategy.

References

1. Bos, J., Nissim, M.: Cross-Lingual Question Answering by Answer Translation. In: Working Notes for the CLEF 2006 Workshop. (2006)
2. Bowden, M., Olteanu, M., Suriyentrakorn, P., Clark, J., Moldovan, D.: LCC's PowerAnswer at QA@CLEF 2006. In: Working Notes for the CLEF 2006 Workshop. (2006)
3. Brants, T.: TnT - a statistical part-of-speech tagger. In: Proceedings of the Sixth Conference on Applied Natural Language Processing (ANLP-2000), Seattle, WA (2000)
4. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: A framework and graphical development environment for robust NLP tools and applications. In: Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics. (2002)
5. Ferrandez, S., Lopez-Moreno, P., Roger, S., Ferrandez, A., Peral, J., Alvarado, X., Noguera, E., Llopis, F.: AliQAn and BRILI QA Systems at CLEF 2006. In: Working Notes for the CLEF 2006 Workshop. (2006)
6. Giampiccolo, D., Forner, P., Penas, A., Ayache, C., Cristea, D., Jijkoun, V., Osenova, P., Rocha, P., Sacaleanu, B., Sutcliffe, R.: Overview of the CLEF 2007 Multilingual Question Answering Track. In: Working Notes for the CLEF 2007 Workshop. (2007)
7. Harabagiu, S., Moldovan, D.: Question Answering. In Mitkov, R., ed.: Oxford Handbook of Computational Linguistics. Oxford University Press (2003) 560 – 582
8. Jijkoun, V., Mishne, G., de Rijke, M., Schlobach, S., Ahn, D., Muller, K.: The University of Amsterdam at QA@CLEF2004. In: Working Notes for the CLEF 2004 Workshop. (2004)
9. LUCENE. (http://lucene.apache.org/java/docs/)
10. Magnini, B., Giampiccolo, D., Forner, P., Ayache, C., Osenova, P., Peas, A., Jijkoun, V., Sacaleanu, B., Rocha, P., Sutcliffe, R.: Overview of the CLEF 2006 Multilingual Question Answering Track. In: Working Notes for the CLEF 2006 Workshop. (2006)
11. Puscasu, G.: A Framework for Temporal Resolution. In: Proceedings of the 4th Conference on Language Resources and Evaluation (LREC2004). (2004)
12. Puscasu, G., Iftene, A., Pistol, I., Trandabat, D., Tufis, D., Ceausu, A., Stefanescu, D., Ion, R., Orasan, C., Dornescu, I., Moruz, A., Cristea, D.: Cross-Lingual Romanian to English Question Answering at CLEF 2006. In: Working Notes for the CLEF 2006 Workshop. (2006)
13. Sutcliffe, R., Mulcahy, M., Gabbay, I., O'Gorman, A., White, K., Slattery, D.: Cross-Language French-English Question Answering using the DLT System at CLEF 2005. In: Working Notes for the CLEF 2005 Workshop. (2005)
14. Tanev, H., Kouylekov, M., Magnini, B., Negri, M., Simov, K.I.: Exploiting Linguistic Indices and Syntactic Structures for Multilingual Question Answering: ITC-irst at CLEF 2005. In: Working Notes for the CLEF 2005 Workshop. (2005)
15. Tapanainen, P., Jaervinen, T.: A Non–Projective Dependency Parser. In: Proceedings of the 5th Conference of Applied Natural Language Processing, ACL. (1997)
16. Tufis, D.: Using a Large Set of EAGLES-compliant Morpho-Syntactic Descriptors as a Tagset for Probabilistic Tagging. In: Proceedings of the Second International Conference on Language Resources and Evaluation. (2000) 1105 – 1112
17. Tufis, D., Cristea, D., Stamou, S.: BalkaNet: Aims, Methods, Results and Perspectives. A General Overview. In Tufis, D., ed.: Romanian Journal on Information Science and Technology. Special Issue on BalkaNet. Romanian Academy (2004)