Towards a Multi-Stream
Question Answering-As-XML-Retrieval Strategy

David Ahn   Sisay Fissaha   Valentin Jijkoun
Karin Müller   Maarten de Rijke   Erik Tjong Kim Sang

ISLA, University of Amsterdam
http://ilps.science.uva.nl/

Abstract: We describe our participation in the TREC 2005 Question Answering track; our main focus this year was on improving our multi-stream approach to question answering and on making a first step towards a question answering-as-XML retrieval strategy. We provide a detailed account of the ideas underlying our approaches to the QA task, report on our results, and give a summary of our findings.

1   Introduction

For the TREC 2005 Question Answering track, we made two major adaptations to the multi-stream system with which we participated in previous years [8, 10]: first by enabling a so-called table stream to process additional question types and generate more candidate answers, and second, and by encoding the document collection and its linguistic annotation in XML thus enabling a QA-as-XML-retrieval strategy. We took part in the main task of the question answering track as well as in the relationship finding task. We describe the results of our participation in the main task in Section 2, and the results of the relationship finding task in Section 3.

2   Main Task

We built on the multi-stream QA architecture that we have been developing over the past few years as part of our work for the TREC and CLEF QA tasks. The architecture has several streams running in parallel: each is based on a different approach to QA and is a self-contained QA system in itself. No new streams were added this year, leaving us with a total of seven streams: a table stream (detailed in §2.1 below), pattern matching and ngram mining (both against the collection and against the web), a Wikipedia stream (which gets answers out of Wikipedia), and Tequesta (which was updated to XQuesta this year, see §2.2 below). For a more detailed description of our multi-stream approach we refer to [1, 2, 5, 7, 9].

Our QA efforts for the main task were concentrated in two areas. First, we enabled the table module to handle more question types and generate more candidate answers (Section 2.1). Second, we upgraded the Tequesta stream by encoding the document collection as well as the different linguistic annotations in XML, thus enabling a "QA-as-XML-retrieval" strategy (2.2). Additionally minor changes were made in the question processing part (2.3) and the named entity recognizer (2.4).

2.1   Table Stream Modifications

An important part of our QA system is a table stream which relies on question-specific tables with answers which were extracted offline rather than during question processing [6]. The tables contain, among others, information on abbreviation expansions, birthdays, country leaders, and event dates. Evaluation of this stream on the 2004 factoid questions showed that while the accuracy of the returned answers was low (28% lenient evaluation), its coverage was even worse (7%). The work described in this section was intended to improve these scores.

The first step we took was to add extra tables. The topics of the tables were chosen based on TREC 2004 factoid questions which the system had been unable to answer: birthplaces of people, definitions, groups and their members, nicknames, and organizations, their founders and founding dates. The tables were filled by applying to the document collection hand-crafted extraction rules which utilized available syntactic and named entity annotation of the documents. Most of the tables were small (about 20k entries or less) with the group table (100k) and the definition tables (5M) as exceptions. The latter grew this large because the extraction rules included rules that derived definitions for arbitrary noun phrases. The new tables enabled the stream to handle previously unanswered questions like *What kind of animal is an agouti?* and their positive effects were larger than the side effect of pattern overgeneration.

Next, the question analysis and the table processing modules were updated. This work included defining new question topics, creating new question templates and making new

links between question topics and tables. We also added a filter to the output of the table stream to make sure that when named entity answers from a certain category (person, location or organization) were requested by the question, ill-typed answers were removed. The filtering step could not be as precise as we would have liked it to be since the categories returned by our named entity analysis are more coarse-grained than the categories of the question analysis.

The adaptations of the table stream had a positive effect on its performance on the TREC 2004 factoid questions. The recall of the top answers went up from 7% (lenient evaluation) to 21% and even the precision of the answers went up (from 28% to 35%).

2.2 Semi-Structured Information Retrieval

The system used in our previous participations in TREC-QA [1, 9] contained a text retrieval stream named Tequesta. This year we changed the task of the stream to XML element retrieval. The document collection was enriched automatically with token boundaries, syntactic and named entity annotation. Both the annotation and the original documents were stored in stand-off XML format. Our new stream XQuesta is able to query the collection with XPath and to retrieve elements that satisfy lexical, syntactic and named entity constraints. For this purpose the document collection was divided in non-overlapping sequences of paragraphs containing at least 400 characters. We found that having access to the corpus annotation improved the quality of the text snippets and allowed more elaborate answer filtering.

2.3 Question Processing

Question processing is the first stage in our system architecture which is common to all the streams. Each of the questions is tagged, and a question class is assigned based on our question classification module. The question types correspond to WordNet words and their senses. For instance, the question type of (1) is COACH%1. To determine the expected answer types, we also use the WordNet hierarchy. In our example, the question type COACH%1 is mapped to the expected answer type PERSON.

(1)   *Q69.6: Who was the coach of the French team?*

Our question analysis was extended by exploiting the syntactic structure of the questions. Thus, we parse the questions using Charniak's parser [3]. The parses provide information about the NP/PP-chunks which are used to determine the focus of the sentence, here *French team*.

2.4 Named Entity Recognition

For factoid questions, we highly depend on the correct output of our named entity recognizer. We discovered some problems when assigning the correct named entities to a sentence. One improvement was to post-process the output of the recognizer by correcting obvious inconsistencies of named entity sequences. The following two examples exemplify the sort of errors corrected in the output. The named entity recognizer often failed to assign the correct tag to names which are included in the name of an organization such as in (2).

(2)   *the*/O *director*/O *of*/O *the*/O *Rose*/E-PER *Institute*/I-ORG *of*/I-ORG *State*/I-ORG *and*/I-ORG *Local*/I-ORG *Government*/I-ORG

In such cases, the named entity tags are changed to the most common one. Moreover, film titles and quotes in quotation marks are hard to detect for the named entity recognizer such as in (3). They are often misclassified as ORG or PER instead of MISC.

(3)   *you*/O *'re*/O *in*/O *"*/O *The*/I-ORG *Sixth*/I-ORG *Sense*/E-ORG *.*/O *"*/O

These errors have also been corrected with a postprocessing filter.

2.5 Handling List and "Other" Questions

The system changes described in the previous sections dealt with factoid questions. For list questions we only made a small modification: we return the same number of answers for each question (eight, when available) because with that number we obtained the best results for the TREC-2004 questions.

The basic strategy for answering the "other" questions has not changed significantly from last year. The method uses IR and NLP techniques to locate documents containing information about the topic, and extract nuggets from the retrieved documents. The nuggets are assigned an initial score, i.e., the retrieval score of the document from which the nugget was extracted. Duplicate or near duplicate nuggets are removed by using a word overlap similarity.

Two approaches are adopted for ranking the nuggets. The first approach makes use of a reference corpus, an encyclopedia, in order to rank the nuggets extracted from the target corpus in case the topic is found in the encyclopedia. Especially, the enyclopedia entry for the topic is extracted and its content is split into sentences. The word overlap score is computed between each nugget and the sentences of the encyclopedia. The nugget is assigned the score for the most similar encyclopedia sentence. Finally, the nuggets are sorted by their respective scores and the top N nuggets are returned.

The second approach is applied to topics which do not have an entry in the encyclopedia. This approach uses centroid-based summarization technique in order to determine the importance of a nugget. This involves computing

centroids, a set of statistically significant words which describe the list of nuggets extracted from the documents. The nuggets are then ranked based on their distance from the centroid [15] and the top N nuggets are returned.

2.6   Runs

We submitted three runs for TREC-QA 2005. We were interested in two research questions. First, would the system perform better with all six[1] streams or with a subset of these streams? This question was important since our work this year has focused on two streams (table and XQuesta) while other streams were not changed. In order to determine the best combination, we evaluated different stream combinations on the TREC 2004 questions. We found that the combination of table, XQuesta, Wikipedia and web ngrams was the best for factoid questions while XQuesta, ngrams from the web and ngrams from the collection performed best for list questions.

Using ngrams from the web for generating factoid answers has as a disadvantage that answers will be generated for almost all questions. This means that few NIL answers will be produced. We considered the presence of NIL answers as an interesting difference between the run with all streams (uams05all) and the run with a subset (uams05be3) and therefore we excluded the web ngrams stream from the factoid questions run. This means that the stream subset run used combinations of three streams: table, XQuesta and Wikipedia for factoid questions and XQuesta, ngrams from the web and ngrams from the collection for list questions. All runs contained the same answers for other questions.

The second question in which we were interested was: Will answer reranking based on web frequencies improve the quality of the top answers? In order to test this, we created a run (uams05rnk) in which the answers of the complete system had been reranked based on their frequency. We replicated the *search engine corroboration* method of the Bangor entry of TREC 2003 [4], in which answers are ranked according to their frequency of occurrence in the summaries of the top 1000 hits returned by a search engine for a query based on the question. We depart from the Bangor method in two respects: first, we use Yahoo rather than Google, because of the more convenient API, and second, we use a different method to construct queries on the basis of questions. Instead of extracting all NPs and VPs from a question to use as a single query, we submit two queries for each question and use the top 500 hits from each. One query is simply the question itself as a set of keywords, i.e., not constrained to be a phrase. The other query consists of *question selectors*, words from the question that are highly likely to occur in a correct answer snippet [16]. Question selectors are extracted from a question using a C4.5 decision tree trained on pairs of questions and correct answer snippets from previous edi-

tions of the TREC QA track; we followed [16] in our training procedure.

2.7   Results

Table 1 gives the combined results for the 3 QA tasks (accuracy for factoids, F score for list and other questions) and the overall scores of our three runs uams05all, uams05be3 and uams05rnk. The column factoid accuracy contains three numbers: exact answers, unsupported answers and inexact answers.

run	factoid accuracy (exact,unsup.,inex.)	list F	other F	overall
be3	0.119 , 0.052 , 0.050	0.064	0.201	0.127
all	0.105 , 0.058 , 0.086	0.050	0.200	0.113
rnk	0.066 , 0.025 , 0.039	0.029	0.201	0.090

Table 1: Results for the main QA task.

The scores for "other" questions are excellent (ranked eighth overall) but the factoid scores (median score of the 71 participating runs was 0.152) are disappointing, especially given the fact that a large part of the work on our system this year was aimed at improving the performance on factoid questions. The answers produced by the limited version of our system (uams05be3) proved to be better than the answers of the complete system (uams05all). Re-ranking based on web-frequencies (uams05rnk) does not improve performance—in fact, it produces substantially fewer correct answers. The primary reason for this decline is that re-ranking tends to prefer answers that are shorter and more common on the web (irrespective of the question). An analysis of the 287 factoid questions for which the uams05rnk run yielded a different answer than the uams05all run reveals that in 241 cases, the answer chosen by re-ranking is more common than the answer chosen without re-ranking (according to Yahoo).[2] This analysis suggests that the first step to improving re-ranking is to use a more sophisticated scoring mechanism that normalizes with respect to the overall frequency of candidate answers; see [14, 17] for discussion of using web statistics in QA.

A potential cause for the low factoid scores could be a ranking problem: correct answers might not be ranked as number one. In order to check this, we estimated the accuracy of the system on factoid questions while looking at the top-n answers rather than only examining the top answer. This evaluation was performed automatically and therefore inexact and unsupported answers have also been counted as correct unlike in the official TREC-QA evaluation where only supported exact answers are correct. As Table 2 shows, our system potentially could have answered close to 60%

---

[1] The web pattern match stream which was employed in the last two editions was not included this year because of technical difficulties.

[2] Of the 46 times in which the answer chosen by re-ranking is less common, that answer is correct (or inexact) 9 times, while the answer chosen without re-ranking is correct (or inexact) 10 times.

of the factoid questions correctly (corresponding to an estimated 32% exact score) with a perfect ranking scheme.

n-answers	uams05all	uams05be3	uams05rnk
top 1	102 (28.2%)	85 (23.5%)	47 (18.0%)
top 2	126 (34.8%)	102 (28.2%)	73 (20.2%)
top 3	139 (38.4%)	109 (30.1%)	90 (24.9%)
top 5	160 (44.2%)	126 (34.8%)	109 (30.1%)
top 10	177 (48.9%)	149 (41.2%)	151 (41.7%)
top 20	190 (52.5%)	163 (45.0%)	188 (51.9%)
any rank	216 (59.7%)	188 (51.9%)	215 (59.4%)
MRR	35.1%	28.8%	21.9%

Table 2: Potential improvements of QA factoid scores.

A close look at the top 1 factoid answers generated for the first five targets by our best run (uams05be3) revealed that the errors made by the system had causes in different modules. Of the 27 factoid questions in this group, 22 were answered incorrectly. Incorrect handling of the target topic or the questions caused errors in twelve of the latter questions. We use Wikipedia for finding the most common version of names in the topic but unfortunately this process mapped the topic *France wins World Cup in soccer* to *FIFA Beach Soccer World Cup* which made finding correct answers for the related six questions hard. In other cases it was just difficult to determine the question topic or to find the right focus words. Question analysis is over-represented in the error cause list but to its defense it should be noted that where input processing failed, problems in other modules usually did not have a chance to surface.

The most important other problem was the justification of Wikipedia answers. In five cases the presented justification document was irrelevant for the question topic. Incorrect named entity labeling also caused problems for five questions although in two of these a solution would require annotation at a micro level which is beyond our current automatic annotation efforts (stadium and placeInItaly rather than location). Another system task which we need to review carefully is answer tiling (i.e., the combination of several partial answers to produce the final answer delivered as output). Four questions displayed tiling problems, often because correct answers were lost after they were combined with incorrect ones.

The three streams involved in this evaluation caused fewer errors than the previously mentioned parts. The Wikipedia and the XQuesta stream each produced three incorrect answers while the table stream generated one of these. The internal ranking of the Wikipedia stream should be improved as should answer filtering within XQuesta. An extra correct answer was missed because the word *competitor* was not linked to its synonym *contestant*. A more careful future use of WordNet could be helpful.

3 Relationship Finding Task

In the Relationship Finding task, systems were given topics, i.e., relatively verbose descriptions of user information needs, and had to return collection nuggets answering these needs. In most cases, a topic set a context and asked an explicit question about relationship between two or more entities. E.g.,

(4) *The analyst is interested in information regarding the Nobel Prize winners from previous years. Records indicate that David Trimble and John Hume shared the Nobel Peace Prize in 1998. Who are Trimble and Hume, and what was their relationship?*

This year we took part in this task with a system based on passage retrieval, word similarity, and named entity matching. The system first retrieved passages relevant to a topic, then extracted sentences from the retrieved passages, and reranked the sentences based on similarity to the topic. We describe the process in some detail (sections 3.1-3.4) and present the results (3.5).

3.1 Passage Retrieval

The collection documents were split into passages of 400 characters (extended to the end of a paragraph). As in the main QA task, we used Lucene [13] with the standard Lnu.ltc model for passage retrieval. We used original full topics as retrieval queries, after (automatically) removing phrases and words likely to be irrelevant for user information needs, such as "*The analyst is interested in information regarding*" in example 4. The top 10 retrieved passages were split into sentences and processed further.

3.2 Topic Processing

For a topic T, our system took its last sentence t (most often, the question expressing the user's information need) for subsequent processing. We extracted named entities from t using our NE tagger; in case t contained fewer than two named entities, we expanded t with preceding sentences, until it contained at least two NEs. For the example 4, t is "*Who are Trimble and Hume, and what was their relationship?*" and two NEs "*Trimble*" and "*Hume*" are extracted. The text t and the list of extracted named entities were used to rerank sentences obtained in the passage retrieval step.

3.3 Word-based Sentence Score

Each retrieved sentence s was assigned a score based on directed word similarity between s and t. In essence, we summed similarities between each word in t and its most similar word in s, according to a specific word similarity measure [11]. More details on the word-based calculation of similarity can be found in [12].

3.4 NE-based Sentence Score

We combined the word similarity-based score with the score based on the number of shared named entities between s and t. To detect whether two sentences contain common named entities (persons, organizations, locations, miscellaneous entities), we used a dictionary of NE variants created from lists of location-adjective correspondences (e.g., *Europe* and *European*) and redirecting links in Wikipedia (e.g., *William Jefferson Blythe IV* is also known as *Bill Clinton*, and *Burma* is an alternative name for *Myanmar*).

Collection sentences were ranked using the sum of word-based and NE-based scores, duplicates and near duplicates were removed using a simple string distance measure, and the best 5 sentences for each topic were returned as answer nuggets.

3.5 Runs and Results

We submitted two fully automatic runs for the 25 official test topics. The run `uams05l` was created as described above and the run `uams05s` was identical, expect for the fact that the nuggets were shortened by removing all definite and indefinite articles, adjectives and adverbs (other than *first*, *last*, etc.). With the second run, we tried to create answers that were as short as possible (evaluation included a length penalty) without removing important material.

Both runs obtained the F-score of 0.12, with the median over all submitted fully automatic runs being 0.12, the best 0.228 and the worst 0.06.

4 Document Ranking Task

Our multi-stream QA architecture does not rely on an ordered set of documents returned from a preprocessing phase. In order to create the obligatory entry for the document ranking task, we returned the justification documents for each answer set in the same order as the final ranking of the answers. Note that our system associates exactly one justification document with each answer.

run	avg. prec.	prec. at 10	R-prec.
uams05all	0.108	0.170	0.129
uams05rnk	0.094	0.156	0.106
uams05be3	0.071	0.132	0.082

Table 3: Results for the document ranking task

Table 3 lists the results obtained by the three runs submitted to the document ranking task with the scores for average precision, precision at 10 answers and precision at R answers, where R is the number of correct answers found by the human assessors. Since our main interest in the QA task lies with QA and not with IR, we have not taken any separate actions to optimize these scores. However, this change

in combination with adaptations of the named entity annotation, question analysis and the table stream have not lead to an improvement over our 2004 scores. We have identified a number of potential causes on which we will work in the coming year.

5 Conclusions

We described our participation in the TREC 2005 Question Answering track. This year, our work for the Question Answering track was largely motivated by the wish to port one of the streams to a "pure" QA-as-XML-retrieval setting, where the target collection is automatically annotated with linguistic information at indexing time, incoming questions are converted to semistructured queries, and evaluation of these queries gives a ranked list of candidate answers.

Neither this work nor the other recent modifications of our system have brought us the score improvements that we were looking for. Still, we believe that the direction we have taken this year is both promising with respect to future system performance as well as scientifically interesting. Frequent modification of the system in combination with continuous evaluation must lead to better scores in the TREC evaluation. And representing QA as semi-structured XML retrieval makes our work interesting for both the XML and the IR community. We expect that the feedback of these communities will have a positive effect on our QA work.

Acknowledgments

References

[1] D. Ahn, V. Jijkoun, G. Mishne, K. Müller, M. de Rijke, and S. Schlobach. Using Wikipedia in the TREC QA Track. In E. Voorhees and L. Buckland, editors, *The Thirteenth Text Retrieval Conference (TREC 2004)*, 2005.

[2] D. Ahn, V. Jijkoun, K. Müller, M. de Rijke, S. Schlobach, and G. Mishne. Making stone soup: Evaluating a recall-oriented multi-stream question answering stream for Dutch. In C. Peters, P. Clough, G. Jones, J. Gonzalo, M. Kluck, and B. Magnini, editors, *Multilingual Information Access for Text, Speech and Images: Results of the Fifth CLEF Evaluation Campaign*, LNCS 3491, pages 423–434. Springer, 2005.

[3] E. Charniak. A Maximum-Entropy-Inspired Parser. In *Proceedings of NAACL-00*, 2000.

[4] T. Clifton, A. Colquhoun, and W. Teahan. Bangor at TREC 2003: Q&a and genomics tracks. In *The Twelfth Text Retrieval Conference (TREC 2003)*. National Institute for Standards and Technology, 2003.

[5] V. Jijkoun and M. de Rijke. Answer selection in a multi-stream open domain question answering system. In S. McDonald and J. Tait, editors, *Proceedings 26th European Conference on Information Retrieval (ECIR'04)*, volume 2997 of *LNCS*, pages 99–111. Springer, 2004.

[6] V. Jijkoun, G. Mishne, and M. de Rijke. Preprocessing documents to answer Dutch questions. In *Proceedings of BNAIC'03*. Nijmegen, The Netherlands, 2003.

[7] V. Jijkoun, G. Mishne, and M. de Rijke. How frogs built the Berlin Wall. In *Proceedings CLEF2003*, volume LNCS. Springer, 2004.

[8] V. Jijkoun, G. Mishne, M. de Rijke, S. Schlobach, D. Ahn, and K. Müller. The University of Amsterdam at QA@CLEF 2004. In C. Peters and F. Borri, editors, *Working Notes for the CLEF 2004 Workshop*, pages 321–324, 2004.

[9] V. Jijkoun, G. Mishne, C. Monz, M. de Rijke, S. Schlobach, and O. Tsur. The University of Amsterdam at the TREC 2003 Question Answering Track. In *Proceedings TREC 2003*, pages 586–593, 2004.

[10] V. Jijkoun, G. Mishne, C. Monz, M. de Rijke, S. Schlobach, and O. Tsur. The University of Amsterdam at the TREC 2003 question answering track. In E. M. Voorhees and L. P. Buckland, editors, *The Twelfth Text REtrieval Conference (TREC 2003)*, pages 586–593. National Institute for Standards and Technology. NIST Special Publication 500-255, 2004.

[11] D. Lin. An information-theoretic definition of similarity. In *Proceedings of International Conference on Machine Learning*, 1998.

[12] D. Lin. Recognizing textual entailment: Is word similarity enough? In *Lecture Notes in Artificial Intelligence*, 2005. to appear.

[13] Lucene. The Lucene search engine, 2005. http://jakarta.apache.org/lucene/.

[14] B. Magnini, M. Negri, R. Prevete, and H. Tanev. Is it the right answer? Exploiting web redundancy for answer validation. In *Proceedings of ACL 40th Anniversary Meeting (ACL-02)*, pages 425–432, University of Pennsylvania, Philadelphia, 2002.

[15] D. R. Radev, H. Jing, M. Stys, and D. Tam. Centroid-based summarization of multiple documents. *Information Processing and Management*, 40:919–938, 2004.

[16] G. Ramakrishnan, S. Chakrabarti, D. Paranjpe, and P. Bhattacharya. Is question answering an acquired skill? In *Proceedings of the 13th international conference on World Wide Web*, 2004.

[17] S. Schlobach, D. Ahn, M. de Rijke, and V. Jijkoun. Data-driven type checking in open domain question answering. *Journal of Applied Logic*, 2006. To appear.