

Question Answering at TREC
Pre-Internship Report
University of Washington, Ling 590
Bob New
March 25, 2008

Abstract

The Question Answering (QA) track in the Text Retrieval Conferences (TREC) is designed to foster development of techniques for answering questions from an open topic domain. Systems submitted for the track display a wide variety of implementation techniques, but also a central theme in a modular pipelined architecture.

This paper is a Pre-Internship report as part of the CLMA program at the University of Washington.

1. Introduction

Question answering systems attempt to return a direct answer to a question, which contrasts to information retrieval systems that return a list of complete documents that the user then manually reviews to find the needed answer. Starting with TREC-8 (1999), the QA track of the TREC conference has provided a means of comparing question answer system results.

The scope and challenge of the questions contained in the TREC test sets has been expanded over time from simple factoid questions to more complex interactive questions. These question types are described further in Section 2. Most of the systems submitted to the QA track are implemented in a modular pipelined design. Many of the functions in these designs are similar in nature and are described in Section 3. Some highlights of papers submitted to the conference are given in Section 4. Much of the work in the QA track has been designing means for evaluation. Some of the evaluation techniques are discussed in Section 5.

2. TREC Question Types

The questions that are part of the TREC QA test sets have been expanded over time. This is a summary of the core questions types that have been used.

- **Factoids.** The first QA Track in the TREC conference (1999) had 198 fact based, short answer questions (factoids) such as “How many calories are there in a Big Mac?”. In that year, each question had at least one document in the search set that contained the answer. In the 2001 data set, questions did not necessarily have an answer, and systems were required to return NIL if an answer was not found.

In 2006, requirements for factoid answers were expanded by stipulating that the most recent temporally correct answer be returned. So for example a question about who is the president of France, the correct answer is the most recent one found in the corpus.

- **List Questions.** Starting in TREC 2003, list questions were added. These can be thought of as factoid questions that had multiple answers. The answers would likely come from multiple documents in the test set so the system had to assemble the answer appropriately (Dang 2006). An example of a list question is, “Which past and present NFL players have the last name of Johnson?”

- **Definition Questions.** Definition questions ask for more interesting information about a person or thing such as “What is a golden parachute?” (Dang 2006). Definition questions were added to the TREC QA track starting in 2003. In 2006, the definition questions were changed to ‘Other’ questions. In this setup, the system returned an unordered set of strings from target documents that represented information about the entity in question. Responses were supposed to not repeat information in the list of strings.

- **Complex Interactive QA (ciQA).** In 2006, the ciQA task was added. The goal was to push QA development research beyond factoid questions into more complex user domains and also to make the systems more interactive with users (Dang 2006).

3. System Architectures

Several architectures have been proposed for the question answering problem. Perhaps the simplest is from Echihabi (2003) is a system with just two modules: an IR engine that returns candidate sentences from documents and an answer rating module that assigns scores to sentence substrings.

Most system descriptions list more modules than this. This is a summary of some of the tasks that are common in many of the system descriptions:

- **Question type classification.** Fundamentally, each question is looking for a kind of answer. The factoid questions are looking for a date, person, thing, etc. Brill (2001) calls out 7 question types such as who, what or how many. Schlaefer (2007) describes a hierarchy of 154 answer types with 44 top level categories. The hierarchy allows for more or less generality in the desired answer. For example questions such as “Where is...” is less specific than “What city ...”.

- **Question rewrite.** This step in the QA pipeline essentially creates search terms that are submitted to the IR search engine. By creating varied search terms, the IR step will return more text pages to extract the answer from. Hovy (2001) discusses using WordNet to “*expand query terms and place all the expanded terms into a Boolean expression. For example, ‘high school’ is expanded to: ‘(high & school) | (senior & high & school) | (senior & high) | high | highschool’.*”

- **IR Search.** This step uses a standard search engine to find content pages that contain the expanded search terms. Burger (2003) reports taking the top 25 pages in this step. Brill (2001) takes 100 pages for each search term generated in the question rewrite step. Xu (2003) takes the top 1000 pages returned for all search terms.

- **Passage selection.** Given the results from the IR search, next a typical QA system selects passages within each page that contain the potential answers. Jijkoun (2003) uses a 200 byte buffer around the query terms as the starting place for the passages.

- **Answer Extraction, Ranking and Filtering.** From the raw passages, the actual answer candidates are extracted. Brill (2001) uses n-grams as the basis for extraction and ranking, while Moldovan (2002) uses a semantic analysis approach.

4. Design Descriptions

This section lists some interesting design descriptions from various papers submitted to TREC.

Brill 2001. This paper describes a ‘data driven’ approach to the problem. Their claim is that by using the WEB as a projection resource, the system can extrapolate from answers found on the WEB to answers that are found in the TREC corpus. Answers to the TREC questions are required to come from the TREC corpus, but this does not preclude using the WEB as a means of enhancing system results.

The main reason the WEB projection works is found in the question rewrite phase. For the question, “What is relative humidity?”, the system generated rewrites such as “is relative humidity”, “relative is humidity” and “relative humidity is”. By taking a simplistic approach to how the copula is moved through the query phrase, the expensive task of semantic and syntactic analysis is avoided.

Then these rewrite phrases are submitted to the IR engine. This is where the WEB projection is useful. By using the massively redundant information on the WEB to search for a particular term, a match is more likely to be found. Then the results of the WEB search are projected onto the TREC corpus. Essentially this means that additional search terms gleaned from the WEB search are used to expand the original question so that a hit is more likely to occur in the smaller TREC corpus.

Moldovan 2002. This team performed an error analysis of their system to see which module contributes to errors in the output. By tracing data as it proceeds through their pipeline system, they can see where each error result is first caused. For example in their system the module that performs keyword expansion (M5) was responsible for 25.7% of the errors. This module is responsible for expanding the set of words given to the IR engine based on morphological, lexical and semantic types. Another 36.4% of the errors are caused by the module that evaluates at the semantic category of the answer (M3).

These two questions represent more than half of the errors in their results. What is missing from the paper, is an analysis of what the results would have been if these two modules were removed. I.e., M5 is expanding the size of the search passed to the IR engine, and M3 is selecting a semantic category that

the answer must match. If these modules were disconnected from the system, presumably, the results would get worse, but how much worse compared to the number of errors that were attributed to them is not discussed.

Xu 2003. Definition questions are questions such as Who is Colin Powell? or What is mold?. (Voorhees 2003). In the TREC task, the answers are lists of strings, with each string representing one facet of the entity being described.

To meet this requirement, Xu (2003) used several techniques to extract facets. Appositives and Copula constructions were extracted from parse trees of the IR results based on simple rules. Also extracted from parse trees were propositions based on predicate argument structure. For example “<PERSON> was born on <DATE>” was one facet extracted for person targets. Another example of a pattern applied to parsed sentences is “<TERM> ,? (is|was)? also? <RB>? called|named|known+as <NP>”. Here <TERM> represents the entity in question. If a sentence matches this construction, then it is added to the list of candidates.

Once the list of facets was extracted from the target corpus, the list was ranked. Appositives and copulas were given highest rank followed by the proposition argument structures. Within these broad classes the items were sorted with tf-idf similarity to the question.

5. Evaluation

Assessing the results of the TREC QA submissions was performed by human judgments. Each answer was read by two judges and if a disagreement was flagged, then a third judge settled the result.

Voorhees (2001) states that absolute scores are not achievable given the nature of the QA task. If either the judges or the questions are changed, then absolute scores are not comparable. Relative scores within a given test run are comparable assuming the same judges are used for all submissions.

For the factoid questions, the score assigned to a system is the percent of the answers that is judged correct. In 2003, the best score achieved was 0.700. In 2006, the best score was 0.578.

The list questions are scored using precision and recall. For this task, answers in a list were required to

be distinct in order to be counted, redundant answers counted against the submission. Given D as the number of Distinct correct answers in a list for a given question, N as the total Number of answers in the list, S as the size of list of possible answers, then:

$$P = D/N$$

$$R = D/S$$

$$F = 2 * P * R / (P + R)$$

The score for the list submission is the average of F for all the list questions (Voorhees 2003).

6. Conclusion

We have discussed some of the attributes of the question answering track at the TREC conferences. The submissions to this track have demonstrated that there are solutions that provide reasonably accurate results for a question in an open domain.

However, Lin (2007) studied the results from the 2004 and 2005 submissions and compared them with a pure IR system. I.e., just put the question into an IR system and look at the result. They used an n-gram overlap measure instead of human judges to compare the original submissions with the plain IR result. The result they report is not particularly encouraging for QA systems for ‘other’ questions: “Most striking is the observation that the baseline Lucene [IR SYSTEM] run is highly competitive with submitted TREC systems.” (Lin 2007).

7. References

- Brill, E., J. Lin, M. Banko, S. Dumais, A. Ng 2001. Data-Intensive Question Answering. TREC QA 2001.
- Burger, J., 2003. *MITRE's Qanda at TREC-12* TREC 2003.
- Dang, H.T., J. Lin, D. Kelly, 2006. *Overview of the TREC 2006 Question Answering Track*. TREC 2006.
- Dekang Lin and Patrick Pantel. 2001. *Discovery of Inference Rules for Question Answering*. Natural Language Engineering 7(4):343-360.
- Dumais, S., M Banko, E. Brill, J. Lin, A. Ng, 2002. *Web Question Answering: Is More Always Better?* ACM SIGIR 2002, Tampere, Finland
- Echihabi, A., D. Marcu, 2003. *A Noisy-Channel Approach to Question Answering*. ACL 2003, Sapporo, Japan.
- Hovy, E., L. Gerber, U. Hermjakob, M. Junk, C. Lin, 2001. *Question Answering in Webclopedia*. TREC-9

Jijkoun, V., G. Mishne, C. Monz, M. de Rijke, S. Schlobach, O. Tsur, 2003. *The University of Amsterdam at the TREC 2003 Question Answering Track*. TREC 2003

Lin, J., 2007. *Is Question Answering Better Than Information Retrieval?* NAACL-HLT 2007

Moldovan, D., M Pasca, S. Harabagiu, M. Surdeanu, (2002). *Performance Issues and Error Analysis in an Open-Domain Question Answering System*. ACL 2002.

Schlaefer, N., J. Ko, J. Betteridge, G. Sautter, M. Pathak, E. Nyberg 2007. *Semantic Extensions of the Ephyra QA System for TREC 2007*. TREC, 2007.

Voorhees, E., 2003. *Overview of the TREC 2003 Question Answering Track*. TREC 2003.

Voorhees, E., 2001. *Overview of the TREC 2001 Question Answering Track*. TREC 2001.

Xu, J., A. Licuanan, R. Weischedel, (2003). *TREC2003 QA at BBN: Answering Definitional Questions*. TREC 2003