Evaluation for Scenario Question Answering Systems

Matthew W. Bilotti and Eric Nyberg

Language Technologies Institute
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, Pennsylvania 15213 USA
{mbilotti, ehn}@cs.cmu.edu

Abstract

Scenario Question Answering is a relatively new direction in Question Answering (QA) research that presents a number of challenges for evaluation. In this paper, we propose a comprehensive evaluation strategy for Scenario QA, including a methodology for building reusable test collections for Scenario QA and metrics for evaluating system performance over such test collections. Using this methodology, we have built a test collection, which we have made available for public download as a service to the research community. It is our hope that widespread availability of quality evaluation materials fuels research in new approaches to the Scenario QA task.

1. Introduction

Since 1999, the TREC (Text REtrieval Conference) series organized by the US National Institute of Standards and Technology (NIST) has provided a forum for comparative evaluation of Question Answering (QA) technology. The growth of the QA field from a nascent research area within Information Retrieval (IR) to a vibrant field in its own right is at least partially attributable to the availability of quality evaluations for emerging technology.

The availability of standardized evaluation techniques drives development of QA technology. At their regular meetings, QA development teams use automatically generated summary evaluation figures to visualize how system performance is evolving as the development process unfolds. The same mechanism is used for regression testing, to prevent the introduction of bugs, or accidental rollback of fixes or improvements. Additionally, the use of standardized test collections over widely-available corpora, and agreed-upon evaluation metrics facilitates the clear communication of research results throughout the QA research community.

In this paper, we discuss the unique evaluation challenges associated with Scenario QA, a form of Question Answering where the user input can include background information and questions with multiple parts, representing a complex information need. We propose an evaluation strategy and metrics for the Scenario QA task, and present a methodology for building a Scenario QA test collection. We report on a successful application of this process at our site and demonstrate how to evaluate Scenario QA system responses with the test collection we have built. Our test collection is available for public download for research purposes, and constitutes our contribution to evaluation materials for the community at large. As access to quality evaluation for Scenario QA improves, we hope to see an acceleration in research into the Scenario QA task.

2. What is Scenario QA?

The most established QA task, and the task that still receives the most attention from researchers, is known as Factoid QA, since it involves the study of questions that can be answered with short, succinct phrases, such as, “Where was Christopher Columbus born?” The TREC series of QA evaluations has included evaluation of Factoid QA since the beginning, and over the years, has gradually raised the difficulty of the task. TREC evaluations are in no small part responsible for the high level of the state-of-the-art in Factoid QA systems.

The focus of the QA research community is changing with the introduction of new, more difficult types of questions representing more complex information needs. In 2003, NIST introduced a type of question, known as the definition question, as part of the TREC QA track. Definition questions, such as “Who is Andrew Carnegie?”, naturally solicit a response in the form of a short paragraph containing pertinent facts, for example that he was a steel magnate and a philanthropist from Pittsburgh who founded the Carnegie Institute of Technology in 1900, which later became Carnegie-Mellon University.

How a QA system should properly formulate an answer to a definition question is still a subject of great debate among community members. When a great many facts are found regarding the focus of the question, how do you choose which to include in the answer? Some groups advocate information utility measures, computed by user modeling or some other means. The solution advocated by the TREC evaluation was to have some facts identified as ‘vital,’ and others, merely ‘okay,’ by a human assessor. Those not designated are presumed to be irrelevant (Voorhees, 2003).

Scenario QA involves the study of a new type of complex question. These scenario questions can not be answered by simple, succinct phrases and are a superclass of definition questions and the relationship questions introduced in the TREC 2005 Relationship QA subtask.[1] The scenario question example shown below was drawn from the Relationship QA subtask:

> Q14: The analyst is interested in Iraqi oil smuggling. Specifically, is Iraq smuggling oil to other countries, and if so, which countries? In addition, who is behind the Iraqi oil smuggling?

[1] See: http://trec.nist.gov

This question begins with a statement about the general information need, then asks a yes or no question, and requests further information if the answer is yes. Finally, there is a follow-up question. What are the qualities that a good answer to this question should possess?

1. The question asks which countries receive the smuggled oil. If there is evidence for persons or organizations that receive the oil, and their geographic locations can be determined, the system should respond with a list of countries. This, and other forms of simple, yet useful, inference, should be a primary focus of the system.

2. The follow-up question asks what individuals and organizations, etc., are responsible for the smuggling. The system should focus on identifying those within Iraq responsible for illegally exporting the oil, rather than compiling a comprehensive list of buyers. In other words, properly checking the semantic constraints is of paramount importance.

3. The system should be able to generalize. Although the question mentions oil, if sufficient evidence is found to suggest smuggling of petroleum derivatives, or other commodities or equipment related to oil, the system should identify these leads.

This is a tall set of requirements. Some of these properties will be impossible to assess without a user study. Aside from the human effort involved in such an undertaking, and the inherent qualitativeness of the results, a user study requires that there be a finished system to be presented to the users. They could be swayed by interface issues to provide negative feedback on research-grade QA technology.

The evaluation challenge in Scenario-Based QA is to find a way for developers to isolate the QA technology from the complete desktop software package designed for analysts, and perform periodic evaluations of it against standardized test collections over well-known corpora, without the need for manual analysis of the QA system output.

3. State-of-the-art in QA Evaluation

Current approaches to the evaluation of complex questions such as definition, relationship and scenario questions fall into two distinct subcategories: human-in-the-loop evaluations, e.g. TREC, and automatic evaluations.

The TREC 2003-2004 definition question tasks had answer keys built by pooling, supplemented by information discovered during question development. As part of the pooling process, the top-n results from each participating system are combined into a pool of results, with duplicates removed, and are shown to an assessor. Judgment is blind to the system that produced the result, and the rank at which that result was retrieved.

One issue with pooling is that it affords only comparative evaluation among the systems that participated in the evaluation. To be fair, NIST-provided lists of relevant documents for each question were never intended to be used as an absolute evaluation set, but many researchers use them as such for lack of a better evaluation method. The TREC 2005 Relationship QA subtask used a different question development process in which the test collection was made reusable by not relying on pooled documents, but the evaluation process still requires a human to match between system output and the answer key.

Two automatic methods for definition question evaluation have been recently published. Lin and Demner-Fushman (2005) use a scoring script called POURPRE to automatically score definition questions against a manually-prepared answer key. They use ngram co-occurrence statistics to approximate manual scoring by a human. They have shown that system rankings from comparative evaluations of definition question systems scored automatically by POURPRE correlate highly with the actual system rankings that use manual scoring, and so they are challenging the notion that scoring a definition question system requires a human to compensate for differences in vocabulary and syntax, and for paraphrase (Voorhees, 2003).

Marton's Nuggeteer (2006) improves upon the functionality of Pourpre by producing scores that more closely approximate the scores manually generated by human assessors. Nuggeteer automates the task of a human assessor by making an individual judgment for each pair of system response and answer key nugget description as to whether the response matches the description. System scores are calculated using the same formula that the NIST assessors use, so the Nuggeteer scores are guaranteed to be comparable to the official scoring. Nuggeteer also offers confidence intervals for its predictions. In terms of accuracy of system rankings, Nuggeteer is comparable to Pourpre.

4. Predicate-Based Evaluation

We propose a comprehensive evaluation strategy for Scenario QA called Predicate-Based Evaluation (PBE). Our strategy is compatible with existing metrics and can be applied automatically.

4.1. What is a Predicate?

A predicate is an instance of a verb's predicate-argument structure. Predicates are automatically extracted from sentence-level nuggets using a shallow semantic parser called ASSERT (Pradhan et al., 2004) that identifies target verbs and chunks noun phrase arguments prior to attaching the arguments to the target verb using PropBank-style role labels (Kingsbury et al., 2002).

4.2. Why Predicates and not Nuggets?

Nugget-based evaluation has been popular for several years in recent TREC evaluations for definition questions (TRECs 2003 and 2004) and relationship questions (TREC 2005); see Voorhees (2003) and (2005). Since a nugget is simply a string extracted from a document, there is a one-to-one correspondence between a predicate as we have defined it and its enclosing sentence, which can be considered a nugget. Because of this feature, PBE is backward compatible with existing nugget-based evaluation technology and judgments at the level of individual nuggets.

An answer key expressed in predicates rather than nuggets makes it easier to automatically compare system responses against the key. Semantic processing can abstract away

Define Recall (**R**) and Precision (**P**):

$$\mathbf{R} = \frac{r}{R}, \ \mathbf{P} = \frac{r}{N}$$

Where:

$r \equiv$ # relevant facts retrieved
$R \equiv$ # relevant facts in the answer key
$N \equiv$ Total # facts in system response

F-measure, then, is defined as:

$$\mathbf{F}(\beta) = \frac{(\beta^2 + 1) \cdot \mathbf{P} \cdot \mathbf{R}}{\beta^2 \cdot \mathbf{P} + \mathbf{R}}$$

Figure 1: Predicate-based Definition of F-measure

from variations in vocabulary and syntax, allowing unification on a higher level, as in (Van Durme et al., 2003). Difficulties can remain where paraphrase or highly different wording occurs in the answer key or the system response, but current work is investigating event ontologies for Scenario QA that can help mitigate this difference. Automatic predicate-based evaluation can still be a solid lower bound on system performance without the assistance of an ontology, and will likely suffice for comparative evaluation of a group of systems, with or with out ontology assistance.

4.3. Metrics

Familiar evaluation metrics, such as precision, recall and F-measure, a weighted harmonic mean of precision and recall (van Rijsbergen, 1979), can be defined with respect to predicates for the purposes of Scenario QA evaluation (see Figure 1). These precision and recall metrics express true precision and recall, not approximations, when coupled with an answer key in which the judgments can be reasonably assumed to be exhaustive. This type of answer key can be constructed using the process outlined in Section 5.1.

5. Building a Reusable Test Collection

Building a reusable test collection for Scenario QA is a two-step task, but the bulk of the work is spent developing an answer key for each scenario question. Once a document collection is chosen and a set of scenario questions formulated, it is time to develop answer keys.

5.1. Developing Answer Keys

The process of developing answer keys is a distributed manual assessment effort in the form of an Interactive Search and Judgment (ISJ) task, in which individual assessors not only judge relevance of documents retrieved, but also formulate the queries used to retrieve those documents. Figure 2 gives a graphical overview of the answer key development process to which the reader can refer throughout this section.

Assessors are recruited from the general community and are asked to self-select on the basis of the following criteria: assessors should be fluent in the language of the scenario questions and document collection, should be comfortable working with a keyword search engine, and should neither be experts in subject domain of the document collection, nor in QA and/or IR research. Assessors that certify that they meet these criteria are welcomed into the program and are offered reasonable hourly compensation for time spent judging documents.

Prior to starting work, an assessor is given a training session in which the task is explained and all the features of the assessment interface are demonstrated. While the assessor is working, the experimenter assigns him or her one scenario question at a time. The choice of which question to tackle next is left up to the experimenter, who may need to balance question topics, or assign some number of questions for multiple assessment, in a way that an automatic question selection mechanism would not be able to handle.

When an assessor begins a new question, he or she is first presented with a keyword query interface designed to look and feel as much like a commercial web-based search engine as is possible. The interface clearly displays the current question and, below that, a field where the assessor types queries. Clicking a button marked “Go” queries the underlying retrieval system and brings up a ranked list of documents, complete with preview ‘snippets’ inspired by popular search engines. The preview snippet is the best-match passage in each document containing the most keyword occurrences. At this point, the assessor can scan the ranked list and choose a document to read, but is free to issue another query at any time.

Once an assessor chooses a document to read, he or she is required to judge it relevant or not relevant to the question. The assessment interface includes some features to make this task easier, including user-configurable keyword highlighting and a ‘Ctrl+F’ find functionality similar to that offered by a standard web browser. Assessors are cautioned that a concentration of highlighted keywords does not constitute an answer, nor does the lack of highlights in a particular passage imply that there is no answer there. This warning is given to assessors in an attempt to encourage them to read more closely rather than simply scanning for highlighted keywords, minimizing judgment errors due to assessors not finding phrasings of the answer that they expect.

When an assessor determines that a document is relevant, he or she is asked to use the assessment interface to draw a box around one or more passages of text containing relevant information. Assessors are told that it takes more time to judge a document not relevant than it it does to judge a document relevant, and that a document is to be considered not relevant unless some relevant information is found and boxed. It is the absence of a boxable region of relevant text that defines a document to be not relevant. When an assessor judges a document not relevant, the interface returns to the ranked list of results, allowing assessors to call up other documents from the list or issue new queries to retrieve new lists of results.

Assessors are given comprehensive guidelines as to what constitutes relevant information in documents, and these guidelines are covered in Section 5.2. When an assessor judges a document to be relevant, he or she is taken to another screen that allows individual judgments on all predi-

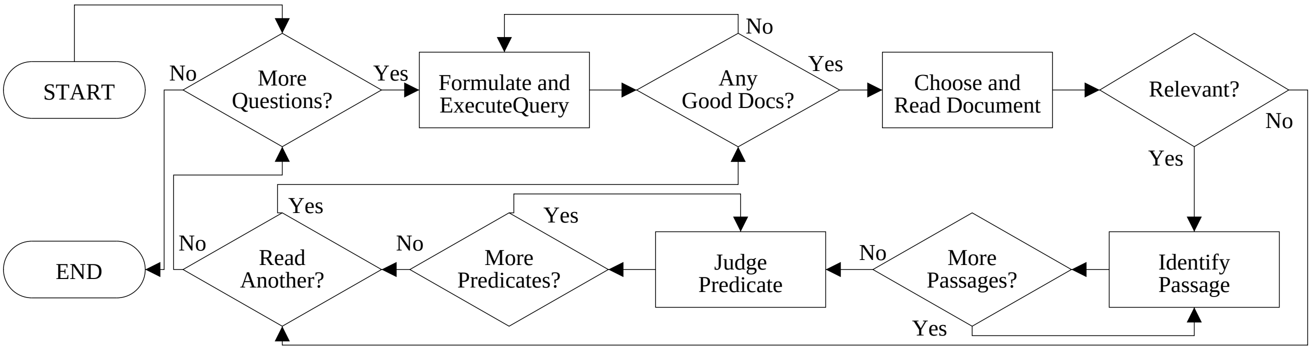

Figure 2: Answer Key Development Process

cates present in the passage. These predicates are extracted by rounding all boxed passages identified by the assessor as containing relevant information to the next sentence boundary, and then running each of these sentences through the ASSERT semantic parser. The output of the parser is shown to the assessor in an abstracted form; each sentence is shown with the target verb highlighted, but arguments are not identified. Assessors understand the language and are capable of visualizing the attachment of arguments to the target verbs of predicates, and it is easier for them to understand when they are told to identify each verb as relevant or not relevant within its individual sentential context. This process has the effect of weeding out rhetorical constructions and predicates centered around matrix clause verbs such as 'seem' and 'believe,' which may not be necessary in the assessor's view to assign relevance to the document.

The assessment interface collects positive and negative judgments at the level of each individual document viewed, and judgments at the passage and predicate levels for relevant documents. In addition, metadata such as number of queries executed, query types executed, ranked lists retrieved by each query, and time spent reading each document, etc., as well as the transcript of each user's interaction with the system are collected for future study.

The methodology presented here is an extension of that presented in (Bilotti, 2004), (Bilotti et al., 2004) and (Lin and Katz, 2005), in which a test collection containing document relevance judgments over the AQUAINT corpus for 120 Factoid questions drawn from the TREC 2002 question set was developed and made available to the research community.[2]

5.2. Guidelines for Determining Document Relevance

This section contains a synopsis of instructions given to assessors regarding which documents are to be considered relevant in certain borderline situations. Specific instruction were given for certain types of questions that an assessor could encounter.

Definition questions, or questions of the form "Who or what is *x*?" were prevalent in the question set. Assessors were given examples of relevant sentences in which *x* was identified by name and some information about *x* was provided, say, in an appositive construction. Documents that mention *x* in passing or that do not give any information about *x* were not judged relevant. Assessors were also warned to read closely to catch for misspelling of names, especially in the case of Arabic and Hebrew names transliterated into Latin characters.

Relationship questions ask for information about the connection between two entities, *x* and *y*, which could be people, organizations, countries, events or most anything else. The relationship may be explicitly stated, as in the causal relationship question "Who or what made *x* do *y*?", or it can be unspecified, as in "What is the connection between *x* and *y*?" In this latter question, the relationship is not known by the user asking the question. The assessors were cautioned that mentions of *x* and *y* that do not make the relationship clear in the text should not be marked relevant. This happened most often in causal questions where certain documents discussed event *y* and the causative event separately, but did not make the relationship explicit. These instructions were given to assessors to ensure that they did not mark a document relevant just because they saw mention of the causative event (the answer) in a document.

The question collection contained a great many multi-part questions, perhaps the most common type of which was the combination definition-relationship question of the form "Who is *x* and what is his relationship to *y*?" For all multi-part questions, assessors were instructed to mark a document relevant if it answers at least one sub-part of the question. In terms of the combination definition-relationship question, this means that, to be judged relevant, a document must define *x*, elaborate on the relationship between *x* and *y*, or do both. Assessors were told that a document that gives the relationship between *x* and *y* does not have to identify *x* by name if there is a definite, specific reference to *x*. An example of a definite, specific reference is "the President of the United States," which identifies a person unambiguously, at least at the time the document was written. This definition of relevance lends itself naturally to the task of Scenario QA, which involves aggregating evidence found in multiple documents when responding to a question.

6. The Javelin Scenario QA Test Collection

The Javelin Scenario QA Test Collection is the product of the first application of the test collection construction methodology proposed above. It consists of judgments at

[2] See: http://www.umiacs.umd.edu/∼jimmylin/downloads/

SUPPLY(Argentina, Egypt, a 20 MW research reactor)
"Argentina confirmed that it has bid to supply a 20 MW research reactor to Egypt." (4796)

SIGNED(Argentina and Egypt, a 15 year nuclear fuel cell agreement in 1998)
"Argentina and Egypt signed a 15 year nuclear fuel cycle agreement in 1988." (4796)

PRODUCE(Egypt and Argentina, six kilograms of plutonium)
BUILD(Egypt and Argentina, a nuclear bomb)
"The CIA (US) is investigating a joint project of Egypt and Argentina to produce six kilograms of plutonium, enough to build a nuclear bomb" (4796)

IMPORT(the Egyptian president, such reactors, from the PRC)
COOPERATE(the Canadian Atomic Energy Commission, with Egypt, in drawing up blueprints for a 600 MW Candu reactor)
PRODUCE(The Commission, nuclear fuel, at Inshas)
"The Egyptian president recently announced his country's intention to import such reactors from the PRC. ... Canada announced the Canadian Atomic Energy Commission would cooperate with Egypt in drawing up blueprints for a 600 MW Candu reactor. The Commission will also work on a project to produce nuclear fuel at Inshas." (8437)

Figure 3: Answer Key for Egyptian Nuclear Reactors Question. Source Document Number from the Collection is Given in Parentheses.

1. CONSTRUCT(Egypt, Pakistan, Iraq and Argentina, to construct a plutonium-producing reactor for nuclear weapons)
"Egypt reportedly cooperates with Pakistan, Iraq, and Argentina to construct a plutonium-producing reactor for nuclear weapons." (35826)

2. CONSTRUCT(Israel, a third nuclear reactor, near the Egyptian border)
"According to Egyptian Atomic Energy Agency specialist Muhammad Mustafa, Israel is making preparations to construct a third nuclear reactor 25km from the town of 'Awajah in Sinai and near the Egyptian border." (15254)

3. BID(Argentina, supply a 10 MW research reactor to Egypt)
"Argentina confirmed that it has bid to supply a 20 MW research reactor to Egypt." (4796)

4. BUILD(Egypt and Argentina, a nuclear bomb)
"The CIA (US) is investigating a joint project of Egypt and Argentina to produce six kilograms of plutonium, enough to build a nuclear bomb" (4796)

5. FINANCE(American Import-Export Bank, the construction of an Egyptian nuclear reactor.)
The American Import-Export Bank pledged in principle to help finance the construction of an Egyptian nuclear reactor. (229)

Figure 4: Hypothetical System Response to Egyptian Nuclear Reactors Question

the document level and at the passage level, in addition to judgments at the level of individual predicates present in the document collection, which in this case is a collection of 39,100 documents from the Center for Nonproliferation Studies known as the CNS Corpus. In total, there are 7548 predicate-level judgments, 1534 passage-level judgments and 1460 document-level judgments for a collection of 199 scenario questions. The questions, formulated with the help of a domain expert, focus on issues related to the proliferation of weapons of mass destruction. The test collection has been released publicly, and is available on the author's web page[3].

The remainder of this section will carry out an example of evaluating a hypothetical response to a scenario question drawn from our test collection. Our example question will be:

> Q175: What efforts to construct nuclear reactors has Egypt made?

Figure 3 shows the answer key that our assessors developed for this question. In the interest of brevity, several of the most illustrative predicates found by our assessors are shown. In the actual test collection, assessors for this question found 31 relevant predicates out of 46 contained in 4 relevant passages of 2 relevant documents.

Figure 4 shows a hypothetical system response to this question. The first, third and fourth-ranked predicates returned are clearly relevant to the question, but only the fourth-ranked predicate appears verbatim in our answer key. Depending on the accuracy of our predicate unification technology, we can match the third-ranked predicate to the first predicate in our answer key. It is a simple inference to make the connection between Argentina bidding to supply Egypt with a reactor and the actual act of supplying it to Egypt, perhaps with some discount factor to express the fact that, at the time the text was written, the supply event had not already taken place.

The first-ranked predicate in Figure 4 is relevant, but does not actually occur in the answer key. This can be blamed on a lack of coverage in the answer key, which undoubtedly exists for some questions. The second and fifth-ranked predicate are not relevant. The second-ranked predicate is an example of a system searching for predicates containing Egypt and nuclear reactor and failing to properly check the directionality of the relation. Here, Israel is the agent of the event corresponding to the construction of the nuclear

[3] See: http://www.cs.cmu.edu/∼mbilotti/resources

reactor, and Egypt occurs in a locative argument. The fifth-ranked predicate discusses financing of nuclear reactors in Egypt, and, while this may be part of the overall picture a good Scenario QA system would present to the information analyst, it does not appear in the answer key because the assessor did not view securing financing as necessarily corresponding to a reactor construction effort despite the fact that there is some relationship between the two events.

Using the metrics defined in Figure 1, we can score this system's response in terms of precision and recall. Given that two of the five predicates retrieved are in the answer key, precision in this case is $\mathbf{P} = 2/5 = 0.4000$. The system only retrieved two of seven relevant facts, so recall is computed as $\mathbf{R} = 2/7 = 0.2857$. F-measure can be computed as well: $\mathbf{F}(1) = 0.3332$, and alternatively $\mathbf{F}(5) = 0.2888$ and $\mathbf{F}(3) = 0.2941$, as was used in TREC 2003 and TREC 2004, respectively. From here, we are free to micro-average over relevant facts in the answer keys, or macro-average over questions to present summary evaluation figures for our Scenario QA system.

7. Ongoing Work

In order to make predicate-based evaluation automatic, it is necessary to have quality predicate-matching techniques. The current state-of-the-art in automatic predicate matching is crude, but ongoing work promises to improve accuracy. The most important next step is to incorporate domain models and ontologies into the predicate matching system such that lexical predicate target verbs can be canonicalized into the (perhaps domain specific) events they encode. Ontologies can also help in the matching of arguments; when a system retrieves a predicate in which a specific argument is a subtype of the argument called for by the answer key, the ontology can help unify the system response and the answer key. Even if ontology-assisted predicate unification is realized, there can still be some gaps in the ontology's coverage. A potential solution to this would be to incorporate recent advances in ngram-based automatic matching of nugget lists to answer keys. Once the structure is matched, if there is an argument that can not be tied to an ontology, it could be a reasonable approximation to use these techniques to check the degree to which that argument matches the answer key.

The Javelin Scenario QA Test Collection currently suffers from a lack of coverage in terms of document-level relevance judgments. In this situation, it is not useful for evaluating the document retrieval component of a Scenario QA system on the basis of the ranked lists of documents it retrieves, independently from the end-to-end system. It is not possible to compute precision and recall because too many of the documents in the ranked list have not been judged. In practice, the *ad hoc* retrieval community builds test collections through a combination of ISJ and pooling. Following their example, we have recently launched an assessment of document pools retrieved by several variants of the retrieval component of our Scenario QA system. Augmenting our test collection with these judgments will allow us to do independent evaluations of our retrieval technology similar to those favored by the *ad hoc* retrieval community.

8. Contributions

In this paper, we have identified a need for new evaluation techniques for Scenario QA. We have defined an evaluation methodology for Scenario QA, and have proposed a process for building Scenario QA answer keys. We have successfully applied this process to develop a complete Scenario QA test collection, consisting of questions and answer keys. The collection is amenable to the use of automatic scoring technology to measure QA system performance, and is compatible with standard evaluation metrics. We are contributing this test collection to the research community at large in the hopes that the availability of quality evaluation technologies spurs growth in Scenario QA research.

9. Acknowledgments

This work was supported in part by Advanced Question Answering for Intelligence (AQUAINT) program award number NBCHC040164.

10. References

M. Bilotti, B. Katz, and J. Lin. 2004. What works better for question answering: Stemming or morphological query expansion? In *Proceedings of the Information Retrieval for Question Answering (IR4QA) Workshop at SIGIR 2004*.

M. Bilotti. 2004. Query expansion techniques for question answering. Master's thesis, Massachusetts Institute of Technology.

P. Kingsbury, M. Palmer, and M. Marcus. 2002. Adding semantic annotation to the penn treebank.

J. Lin and D. Demner-Fushman. 2005. Automatically evaluating answers to definition questions. In *Proceedings of the 2005 Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*.

J. Lin and B. Katz. 2005. Building a reusable test collection for question answering. *Journal of the American Society for Information Science and Technology*. (in press).

G. Marton. 2006. Nuggeteer: Automatic nugget-based evaluation using descriptions and judgements. MIT CSAIL Work Product 1721.1/30604 http://hdl.handle.net/1721.1/30604.

S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky. 2004. Shallow semantic parsing using support vector machines.

B. Van Durme, Y. Huang, A. Kupsc, and E. Nyberg. 2003. Towards light semantic processings for question answering. In *Proceedings of HLT/NAACL 2003 Workshop on Text Meaning*.

C. van Rijsbergen. 1979. *Information Retireval*. Butterworths, London.

E. Voorhees. 2003. Overview of the trec 2003 question answering track. In *Proceedings of the 12th Text REtrieval Conference, November 2003 (TREC 2003)*.

E. Voorhees. 2005. Trec 2005 question answering track guidelines. In *Proceedings of the 14th Text REtrieval Conference, November 2005 (TREC 2005)*.