Building a Question Answering Test Collection

Ellen M. Voorhees, Dawn M. Tice
National Institute of Standards and Technology
100 Bureau Drive, STOP 8940
Gaithersburg, MD 20899-8940
{ellen.voorhees,dawn.tice}@nist.gov

Abstract

The TREC-8 Question Answering (QA) Track was the first large-scale evaluation of domain-independent question answering systems. In addition to fostering research on the QA task, the track was used to investigate whether the evaluation methodology used for document retrieval is appropriate for a different natural language processing task. As with document relevance judging, assessors had legitimate differences of opinions as to whether a response actually answers a question, but comparative evaluation of QA systems was stable despite these differences. Creating a reusable QA test collection is fundamentally more difficult than creating a document retrieval test collection since the QA task has no equivalent to document identifiers.

1 Introduction

The Text REtrieval Conference (TREC) is a series of workshops organized by the National Institute of Standards and Technology (NIST) and designed to advance the state-of-the-art in information retrieval (IR) [15]. The workshops have focused primarily on the traditional IR problem of retrieving a ranked list of documents in response to a statement of information need. However, in many cases a user has a specific question and would much prefer that the system return the answer itself rather than a list of documents that contain the answer. To address this need, the TREC-8 workshop sponsored the Question Answering (QA) Track that focused on the problem of retrieving answers rather than document lists.

The primary goal of the QA track was to foster research on the QA problem and to document the current state-of-the-art. NIST had an additional goal for the track as well: to explore how best to evaluate QA systems. Text retrieval researchers have a long history of evaluating their systems using test collections consisting of a set of documents, a set of queries, and *relevance judgments*, a list of the documents that should be returned for each query as determined by human assessors. Indeed, the primary way TREC has been successful in improving text retrieval performance is by creating appropriate test collections for researchers to use when developing their systems. NIST used the QA track to investigate whether the use of human assessors is appropriate for a task other than text retrieval, and whether an equivalent to text retrieval's test collections can be created to support QA system development.

This paper summarizes the findings associated with both track goals. The next section situates the particular task that was performed in the track in the context of previous question answering research. Section 3 explains how the retrieval results were evaluated and gives the evaluation scores for the runs submitted to the track. Since one of the main findings of the track is that assessors' opinions as to the correctness of a response differ, Section 4 explores the effect of these differences on the evaluation. That section also discusses the difficulties of building a true equivalent of a text retrieval test collection for the QA task.

2 Question Answering

"Question answering" covers a broad range of activities from simple yes/no responses for true-false questions to the presentation of complex results synthesized from multiple data sources. Question-answering systems with widely varying capabilities have been developed, depending on different assumptions as to the precise task the system should perform and the resources available to it. In this section we briefly list some of the approaches taken to question answering in the past, and then give a detailed description of the task in the TREC-8 QA track.

2.1 Previous work

The first question-answering computer systems were developed as vehicles for natural language understanding research. For example, one of the earliest computer-based question-answering systems, STUDENT, read and solved high school algebra word problems [18]. Correctly solving the algebra problem was taken as a demonstration that the system understood the written statement of the problem. Since understanding language requires world knowledge, systems were limited by by the amount of knowledge they contained. Thus Winograd's SHRDLU system was constrained to a simple block world [18], while the LUNAR system allowed geologists to ask questions about moon rocks [19]. The LUNAR system is notable in that it was the subject of one of the first user evaluations of question-answering systems. LUNAR was demonstrated at the Second Annual Lunar Science Conference in January 1971, and geologists were encouraged to ask it questions. Of the 111 questions that were within the scope of the moon rock data (and were not comparatives), 78% were answered correctly, 12% failed for "clerical" reasons, and 10% had more serious errors.

Knowledge-intensive question-answering systems continue to be developed as the result of natural language understanding research, but have also been developed to accomplish particular tasks [17]. The LUNAR system was an early example of natural language front-ends to database systems such as Microsoft's English Query[1] (`http://www.microsoft.`

[1] Products are given as examples only. The inclusion or omission of a particular company or product implies neither endorsement nor criticism by NIST.

com/technet/sql/engquer.asp) or ELF Software's Access ELF (http://www.elfsoftware.com/home.htm). Question-answering is also the main method of interacting with expert systems, both to pose the problem to be solved and to view the system's justification for its response. The systems developed within the DARPA High-Performance Knowledge Bases Project (HPKB) are recent examples of systems designed to answer complex questions within a narrow domain [5], though even here different parts of the system place different burdens on the knowledge base. For example, the START system can answer simpler questions using knowledge bases mined from the World Wide Web [7].

In a separate body of QA research, no attempt was made to have systems understand text. Instead, the goal was to extract a small piece of text that answers the user's question from a much larger body of text. These systems do not rely on a knowledge-base, and are therefore domain-independent, but they do depend on the answer being present in the text that is searched. O'Connor described a method for retrieving "answer-passages" at a time that most commercial retrieval systems were returning bibliographic references [11]. The MURAX system used an on-line encyclopedia as a source of answers for *closed-class* questions, which Kupiec defined as "a question stated in natural language, which assumes some definite answer typified by a noun phrase rather than a procedural answer" [8]. The FAQ Finder system used files containing question and answer pairs as developed for human readers of Usenet news groups to answer user's questions [4].

Information extraction (IE) systems—such as those used in the Message Understanding Conferences (MUCs, see http://www.muc.saic.com)—recognize particular kinds of entities and relationships among those entities in running text. While not strictly question-answering systems, the goal of IE systems is to populate database-like tables with the extracted data to facilitate future question answering. Like traditional QA systems, IE systems generally depend on domain knowledge to find appropriate text extracts. However, the trend in IE research has been toward more shallow and less domain-dependent techniques (see, for example, the FASTUS [1] or PLUM [2] systems).

2.2 The TREC-8 QA task

The goal of the TREC-8 QA track was to foster research that would move retrieval systems closer to *information* retrieval as opposed to *document* retrieval. Document retrieval systems' ability to work in any domain was considered an important feature to maintain. At the same time, the technology that had been developed by the information extraction community appeared ready to exploit. Thus the task for the TREC-8 QA track was defined such that both the information retrieval and the information extraction communities could work on a common problem. The task was very similar to the MURAX system's task except that the answers were to be found in a large corpus of documents rather than an encyclopedia. Since the documents consisted mostly of newswire and newspaper articles, the domain was essentially unconstrained. However, only closed-class questions were used, so answers were generally entities familiar to IE systems.

Participants were given a document collection and a test set of questions. The document collection was the TREC-8 ad hoc document collection, which consists of approximately 528,000 articles from the *Los Angeles Times*, the *Financial Times*, the Foreign Broadcast Information Service (FBIS), and the *Federal Register*. The questions were 200 fact-based, short-answer questions such as those given in Figure 1. Each question was guaranteed to have at least one document in the collection that answered it. For each question, participants returned a ranked list of five [*document-id*, *answer-string*] pairs such that each answer string was believed to contain an answer to the question. Answer strings were limited to either 50 or 250 bytes depending on the run type. Human assessors read each string and made a binary decision as to whether or not the string contained an answer to the question in the context provided by the document. Individual questions received a score equal to the reciprocal of the rank at which the first correct response was returned (or 0 if none of the five responses contained a correct answer). The score for a run was the mean of the individual questions' reciprocal ranks.

- How many calories are there in a Big Mac?
- What two US biochemists won the Nobel Prize in medicine in 1992?
- Who was the first American in space?
- who is the voice of Miss Piggy?
- Where is the Taj Mahal?
- What costume designer decided that Michael Jackson should only wear one glove?
- In what year did Joe DiMaggio compile his 56-gam hitting streak?
- What language is commonly used in Bombay?
- How many Grand Slam titles did Bjorn Borg win?
- Who was the 16th President of the United States?

Figure 1: Example questions used in the question answering track.

The final set of 200 questions was chosen from a much larger pool of candidate questions. The candidate questions came from four different sources: TREC QA participants, the NIST TREC team, the NIST assessors, and question logs from the FAQ Finder system. These different sources provided different kinds of questions. The TREC participants and NIST staff have knowledge of question answering technology and could use that knowledge to select "interesting" questions. The assessors have limited technical knowledge, and so represented a general user's point of view. However, the assessors created their questions from the test document collection specifically for the track, and thus their questions do not represent natural information-seeking behavior. The questions taken from the FAQ Finder logs, on the other hand, were submitted to the FAQ Finder system by undergraduate students who were genuinely interested in the answers to the questions[2].

NIST staff filtered the candidate questions to obtain the final set of 200. Many of the FAQ Finder questions did not have answers in the document collection so could not be used. Questions that were extremely obvious back-formations of a document statement were removed, as was any question a staff member thought was fuzzy, ambiguous, or unclear. Most questions

[2]The FAQ Finder question logs were given to NIST by Claire Cardie of Cornell University, with permission of Robin Burke, the creator of the FAQ Finder system.

whose answer was a list were removed, though a few questions that required two responses were retained after making the request for two answers explicit in the question (e.g., *What two US biochemists won the Nobel Prize in medicine in 1992?*). The final test set of questions contained 127 questions originally submitted by participants or NIST staff, 49 questions from the assessors, and 24 questions from the FAQ Finder logs.

Despite the care taken to select questions with straightforward, obvious answers and to ensure that all questions had answers in the document collection, once assessing began it became clear that there is no such thing as a question with an obvious answer. Not only did most questions have more different answers than anticipated, but the assessors determined that two of the 200 questions in fact had no answer in the document collection. The final scores were therefore computed over the remaining 198 questions.

3 Evaluating Retrieval Results

The information retrieval and information extraction communities have different traditions in evaluation practice. In IR, the user (usually as represented by a relevance assessor) is the sole judge of a satisfactory response. This reflects the reality that different people have different opinions about whether or not specific documents are relevant to a question [12], but also makes resulting scores comparative, not absolute. The IE community has traditionally operated on the assumption that a gold standard answer key that enumerates *all* acceptable responses (for example, a MUC template) can be developed by an application expert. System responses must exactly match the answer key, but final scores are absolute. NIST's subgoal for the track was to explore the consequences of using the traditional IR evaluation methodology for a task other than text retrieval. We hypothesized that different people would have conflicting opinions as to what constitutes an acceptable response to a question, making it impossible to construct a single comprehensive answer key. But if such differences were found, the effect of the differences on QA evaluation must then be assessed.

To explore these issues, each question was independently judged by three different assessors. This section describes the training the assessors received and their perception of the judging task, as well as the scores computed for the systems using an adjudicated set of assessments. More details regarding question judging are given in the track report in the TREC-8 proceedings [16].

3.1 Assessor training

Since experience with the document relevance judging task has demonstrated the importance of adequate training for assessors [6], the assessors who did QA judging received training developed specifically for the task. The purpose of the training was to motivate the assessors' task and provide general guidance on the issues that would arise during assessing rather than to drill the assessors on a specific set of assessment rules.

The QA training session lasted approximately two hours and consisted of judging specially-constructed answer pools for four training questions. Before any judging took place, the assessor was given the following instructions.

> Assume there is a user who trusts the answering system completely, and therefore does not require that the system provide justification in its answer strings. Your job is to take each answer string in turn and judge if this answer string alone were returned to the trustful user, would the user be able to get the correct answer to the question from the string.

The training questions introduced the fundamentals of QA judging to the assessors: that the answer strings would contain snippets of text that were not necessarily grammatically correct and might even contain word fragments; that the answer string did not need to contain justification to be counted as correct; that the assessors were to judge the *string* not the document from which the string was drawn; that document context must be taken into account; and that the answer string had to be responsive to the question. Document context was vital for questions whose answers change over time. Responses to questions phrased in the present tense (*Who is the prime minister of Japan?*) were judged as correct or incorrect based in the time of the document associated with the response. Requiring that the answer string be responsive to the question addressed a variety of issues. If answer strings contained multiple entities that were of the same semantic category as the correct answer, but did not indicate which of those entities was the actual answer, the response was judged as incorrect. So, for example, an answer string consisting of a list of names returned for a "who" question was judged as incorrect whether or not the correct name was included on the list. Certain punctuation and units were also required. Thus "5 5 billion" was not an acceptable substitute for "5.5 billion", nor was "500" acceptable when the correct answer was "$500". Finally, unless the question specifically stated otherwise, correct responses for questions about a famous entity had to refer to *the* famous entity and not to imitations, copies, etc. For example, questions 199 and 200 asked for the height of the Matterhorn (i.e., the Alp) and the replica of the Matterhorn at Disneyland, respectively. Correct responses for one of these questions were incorrect for the other.

3.2 Judging the test set

The QA track had 20 participants and a total of 45 runs submitted to it. To judge a question, an assessor was required to judge each answer string that was in that question's answer pool; the answer pool for a question consisted of the distinct [*doc-id*, *answer-string*] pairs that were returned for that question across the 45 runs. The mean size of an answer pool was 191.6 pairs (minimum pool size was 169 pairs, maximum pool size was 207 pairs), and the pools contained a mean of 55.3 distinct documents (min 28, max 93).

The assessors interacted freely with NIST staff during assessing, asking for clarification of the assessment guidelines and verifying their application of the guidelines to particular cases. This interaction provided an informal mechanism for learning how the assessors perceived their task. Two more formal methods for gathering this information were also instituted. For each question, the assessors kept a log of the actual answer(s) to the question and any comments they had about judging it. The most detailed information came from a series of "think-aloud" observations of assessors judging an entire question. During a think-aloud session, the assessor was asked to think aloud as he or she considered each answer string in the answer pool. An observer recorded the comments as the assessor judged the strings. Eight think-aloud sessions were held, one each with five different assessors on five different questions plus all three assessors on a sixth question.

The evidence gathered in these ways suggests that the assessors understood their task and were able to do it. It also confirms the hypothesis that assessors have differences of opinion as to what constitutes an acceptable answer even for these deliberately constrained questions. Two prime examples of where such differences arise are the completeness of names and the granularity of dates and locations. For example, for the question *When did French revolutionaries storm the Bastille?*, some assessors accepted "July 14", others accepted "1789", and everyone accepted "July 14, 1789". Similar issues arise with locations. For the question *Where was Harry Truman born?*, some assessors accepted only Lamar, Missouri, while others accepted just Missouri. No assessor accepted just USA, though for other questions country-only designations were judged as acceptable. People are addressed in a variety of ways as well. The assessor training suggested that surname-only is usually acceptable while first-name-only seldom is. Besides obvious exceptions such as Cher or Madonna, there are the different forms of address in other cultures. The full name of the recipient of the 1991 Nobel Peace Prize is Aung San Suu Kyi. Some assessors accepted all of "Aung San Suu Kyi", "Suu Kyi", "San Suu Kyi", and "Kyi".

Granularity of names, times, and locations were not the only disagreements among assessors. Assumed context also mattered, and differed among assessors. For the question *Where is the Taj Mahal?* one of the three assessors accepted Atlantic City, NJ (home of the Taj Mahal casino) as an acceptable response even in light of the judging guidelines that stated replicas and imitations should not be used as answers for questions about a famous entity. For this assessor, the casino was sufficiently well-known to be an entity in its own right.

On average, 6% of the answer strings that were judged were disagreed on. Looking at the total percentage of answer strings that had disagreements is somewhat misleading, though, since a large percentage of the answer strings are obviously wrong and assessors agree on those. Following the document relevance judgment literature [9], we can compute the *overlap* in the sets of strings that were judged correct. Overlap is defined as the size of the intersection of the sets of strings judged correct divided by the size of the union of the sets of strings judged correct. Thus, an overlap of 1.0 means perfect agreement and an overlap of 0.0 means the sets of strings judged as correct were disjoint. The mean overlap across all three judges for the 193 test questions that had at least 1 correct string found was .641.

Since an acceptable response clearly depends on the question and on the person receiving the answer, there is little hope that more carefully defined assessor instructions or more proscribed question selection procedures will eliminate inconsistencies in judgments among assessors. But this is just as well since forcing agreement among the assessors would defeat the purpose of the evaluation. Eventual end-users of the technology will have different opinions and expectations, and the technology will have to be able to accommodate those differences to be useful.

3.3 Retrieval results

The judgment set used to score the track results was a combination of the three assessors' judgment sets. For each question, the three sets of judgments were compared, and any [*docid*, *answer-string*] pair that had two different judgments was reviewed by an adjudicator. The adjudicator's role was not to provide a fourth judgment, but rather to decide if the differences in judgments were caused by differences of opinions or misapplication of the assessing guidelines. If a difference was a matter of opinion, the judgment of the majority of the assessors was used. If the difference was caused by an incorrect application of the judging guidelines, or caused the judgments to be inconsistent across the set of strings in the pool, the adjudicator overruled the majority opinion.

Each of the runs submitted to the track was evaluated using the mean reciprocal rank and the number of questions for which no answer was found as computed by the adjudicated judgment set. The results for 41 of the runs (the four remaining runs were in error) are given in Table 1 where the first four columns in order are: name of the run, the organization that submitted the run, the mean reciprocal rank score and the number of questions for which no correct response was found. The remaining columns are defined in Section 4.1. The table is split between the 50-byte and the 250-byte runs and is sorted by decreasing mean reciprocal rank within run type.

The most accurate of the systems were able to answer more than 2/3 of the questions. When an answer was found at all, it was likely to be ranked highly. Not surprisingly, allowing 250 bytes in a response is an easier task than limiting responses to 50 bytes: for every organization that submitted runs of both lengths, the 250-byte limit run had a higher mean reciprocal rank. On the other hand, the most accurate result was a 50-byte run. The 250-byte limit appears to be large enough so that traditional text retrieval techniques that find best-matching passages are successful, but the 50-byte limit requires more explicit linguistic processing to extract the correct response.

Most of the systems that attempted more explicit processing used the same general approach. First, the system attempts to classify the question according to the type of its answer. So, for example, a "who" question would entail a person or an organization as an answer; a "when" question would entail a time designation as an answer; "how many" would entail a quantity; etc. Next the system retrieves a relatively small portion of the document collection (500-1000 documents) based on similarity to the question, and performs a shallow parse of those documents. The shallow parse enables the system to detect entities of the types entailed by the question. If an entity of an entailed type is found close to the question's words, the system returns the entity as the response. If no appropriate answer type is found, the system falls back to best-matching-passage techniques.

The systems' lack of true understanding of the text sometimes led to amusing responses. One response to the question *Who was the first American in space?* was Jerry Brown, taken from a document which says "As for Wilson himself, he became a senator by defeating Jerry Brown, who has been called the first American in space." A similar response was returned for the question *Who wrote `Hamlet'?*: "`Hamlet,' directed by Franco Zeffirelli and written by . . . well, you know." (Both responses were judged as incorrect.) Answer strings were usually associated with documents that did contain an answer, but the strings did not necessarily contain that answer. The answer pool for the question *Who is the author of the book, `The Iron Lady: A Biography of Margaret Thatcher'?* contained not only the string "The Iron Lady; A Biography of Margaret Thatcher by" but also responses of Ronald Reagan, Giroux, Deirdre Bair, Alfred A. Knopf, Lady Dorothy Neville, and Samuel Beckett.

4 Evaluating the Question Answering Evaluation

This section explores the validity of the evaluation methodology used in the track. Two questions are addressed: whether QA systems can be meaningfully compared given that assessor opinions differ, and whether the methodology can be used to create a QA test collection.

4.1 The stability of QA evaluation

Since three different judgment sets are available for each question, it is possible to directly measure the effect different judgments have on the scores of the QA runs by using the same procedure as was used to gauge the effect of differences in relevance judgments on document retrieval system evaluation [14]. This procedure quantifies the changes in *system rankings* when different judgment sets are used to score the runs. A system

Run Tag	Organization	MRR	# not found	Mean MRR	σ	Min	Max
textract9908	Cymfony, Inc.	.660	54	.617	.013	.564	.676
SMUNLP1	Southern Methodist U.	.555	63	.504	.014	.446	.557
attqa50e	AT&T Research	.356	109	.355	.008	.323	.384
IBMDR995	IBM	.319	110	.288	.011	.233	.334
xeroxQA8sC	Xerox Research Centre Europe	.317	111	.303	.011	.257	.347
umdqa	U. of Maryland	.298	118	.293	.009	.257	.330
MTR99050	MITRE	.281	118	.257	.010	.217	.303
IBMVS995	IBM	.280	120	.269	.010	.231	.306
nttd8qs1	NTT Data Corp.	.273	121	.257	.009	.211	.293
attqa50p	AT&T Research	.261	121	.218	.011	.174	.262
nttd8qs2	NTT Data	.259	120	.238	.009	.197	.276
CRL50	New Mexico State U.	.220	130	.195	.008	.160	.226
INQ634	U. of Massachusetts	.191	140	.178	.008	.145	.212
CRDBASE050	GE/U. of Pennsylvania	.158	148	.152	.008	.122	.186
INQ638	U. of Massachusetts	.126	158	.123	.006	.098	.146
shefinq50	U. of Sheffield	.081	182	.063	.007	.040	.086
shefatt50	U. of Sheffield	.071	184	.066	.005	.056	.076

a) Runs with a 50-byte limit on the length of the response.

SMUNLP2	Southern Methodist U.	.646	44	.627	.010	.583	.662
attqa250p	AT&T Research	.545	63	.543	.010	.502	.582
GePenn	GE/U. of Pennsylvania	.510	72	.496	.009	.458	.529
attqa250e	AT&T Research	.483	78	.480	.009	.439	.517
uwmt9qa1	MultiText Project	.471	74	.462	.008	.431	.492
mds08q1	Royal Melbourne Inst. Tech	.453	77	.452	.008	.418	.486
xeroxQA8lC	Xerox Research Centre Europe	.453	83	.440	.010	.396	.480
nttd8ql1	NTT Data Corp.	.439	79	.441	.010	.398	.478
MTR99250	MITRE	.434	86	.415	.012	.367	.465
IBMDR992	IBM	.430	89	.429	.008	.394	.459
IBMVS992	IBM	.395	95	.393	.009	.355	.429
INQ635	U. of Massachusetts	.383	95	.369	.010	.326	.408
nttd8ql4	NTT Data Corp.	.371	93	.353	.011	.305	.402
LimsiLC	LIMSI-CNRS	.341	110	.340	.008	.308	.376
INQ639	U. of Massachusetts	.336	104	.330	.009	.295	.365
CRDBASE250	GE/U. of Pennsylvania	.319	111	.310	.008	.277	.343
clr99s	CL Research	.281	115	.273	.009	.238	.309
CRL250	New Mexico State University	.268	122	.250	.009	.211	.290
UIowaQA1	U. of Iowa	.267	117	.270	.007	.246	.298
Scai8QnA	Seoul National U.	.121	154	.114	.006	.089	.137
shefinq250	U. of Sheffield	.111	176	.099	.007	.071	.121
shefatt250	U. of Sheffield	.096	179	.088	.005	.076	.101
NTU99	National Taiwan U.	.087	173	.083	.006	.060	.101
UIowaQA2	U. of Iowa	.060	175	.059	.006	.041	.082

b) Runs with a 250-byte limit on the length of the response.

Table 1: QA retrieval results. The official track scores are the mean reciprocal rank (MRR) and the number of questions for which no answer was found (# not found). Also given are scores computed over 1-judge qrels: the mean and standard deviation (σ) of the MRR over 100,000 qrels, and the minimum and maximum MRR observed over the set of 100,000 qrels.

ranking is a list of the systems under consideration sorted by decreasing mean reciprocal rank as computed by a particular judgment set (abbreviated as a *qrels* below). The distance between two rankings is computed using a correlation measure based on Kendall's tau [13]. Kendall's tau computes the distance between two rankings as the minimum number of pairwise adjacent swaps to turn one ranking into the other. The distance is normalized by the number of items being ranked such that two identical rankings produce a correlation of 1.0, the correlation between a ranking and its perfect inverse is -1.0, and the expected correlation of two rankings chosen at random is 0.0.

Recall that a qrels contains a yes/no judgment for each unique [*doc-id*, *answer-string*] pair across the set of runs being evaluated for each question. Two basic types of qrels can be defined: a *multiple-judge qrels* in which the judgment for each pair is based on some function of the different assessors' judgments, and a *1-judge qrels* where the judgments for all the pairs for one question are one assessor's opinion. The adjudicated qrels used to produce the official track scores is one example of a multiple-judge qrels. We constructed three other multiple-judge qrels, including the majority qrels in which each judgment is the majority opinion of the assessors with no overruling by the adjudicator; the union qrels in which the judgment for a pair is yes if any assessor marked it yes; and the intersection qrels in which the judgment for a pair is yes only if all three assessors marked it yes. In addition to these multiple-judge qrels there are

	Mean τ	Min τ	Max τ
in subsample	.9632 (15.1)	.9171 (34)	.9976 (1)
with adjudicated	.9563 (17.9)	.9146 (35)	.9878 (5)

a) correlations for one-judge rankings

	τ
majority	.9683 (13)
union	.9780 (9)
intersection	.9146 (35)
a 1-judge qrels	.9683 (13)

b) correlations with the adjudicated ranking

Table 2: Kendall correlation (τ) of system rankings and corresponding number of pairwise adjacent swaps produced by different qrels sets. With 41 systems, there is a maximum of 820 possible pairwise adjacent swaps.

3^{198} different 1-judge qrels (3 different judgments for each of 198 questions).

We evaluated each of the 41 runs using a random sample of 100,000 1-judge qrels, and calculated the sample mean and standard deviation of the mean reciprocal rank for each run. We also recorded the minimum and maximum mean reciprocal rank observed for each system over the set of 100,000 qrels. These values are given in the second half of Table 1. As is true for document retrieval evaluation, the absolute values of the mean reciprocal rank scores do change when different qrels are used to evaluate the runs. The largest standard deviation for any run was .014; thus any two runs whose mean reciprocal rank scores are within .014 of each other must be considered to be equivalently accurate. The scores for the majority and adjudicated qrels are between the minimum and maximum 1-judge qrels scores for all runs. The union and intersection scores were frequently outside this range, which is a difference between QA evaluation and document retrieval evaluation.

For comparative evaluations, the important thing is how the relative scores of runs change when different qrels are used. Thus we created system rankings for each of the qrels and computed the Kendall correlations among them. The mean of the Kendall correlations for the 1-judge qrels was computed in two ways. In the first case, we took the mean of all pair-wise correlations in a random sample of 1000 of the 1-judge rankings. In the second case, we took the mean of the Kendall's correlation between the adjudicated qrels and all 100,000 1-judge rankings. We also computed the correlation between the adjudicated ranking and each of the other multiple-judge rankings. The correlations are given in Table 2. The numbers in parentheses show the number of pairwise adjacent swaps a correlation represents given that there are 41 different runs being ranked. Since any two 1-judge qrels are likely to contain exactly the same judgments for 1/3 of the questions on average, the qrels are not independent of one another. Thus the Kendall correlation shown may be slightly higher than it would be with completely independent qrels.

The correlations in the top part of Table 2 show that QA system rankings produced from 1-judge qrels are at least as stable as document retrieval system rankings in the face of changes in judgments. There are minor differences in the rankings, but most of those differences are caused by runs whose mean reciprocal rank scores are very close. This confirms that 1-judge rankings are essentially equivalent to one another for the purpose of comparative evaluation of QA systems. Furthermore, the second half of Table 2 suggests that 1-judge qrels are also equivalent to the expensive adjudicated qrels for comparative evaluations. Thus the QA track evaluation is valid to the extent that the evaluation is stable under changes to the judgments used to produce the scores.

4.2 QA test collections

One of the key goals of the QA track was to build a reusable QA test collection—that is, to devise a means to evaluate a QA run that uses the same document and question sets but was not among the runs judged by the assessors. The qrels sets described above do not constitute a reusable test collection because the unit that is judged is the entire answer string. Different QA runs very seldom return exactly the same answer strings, and it is quite difficult to determine automatically whether the difference between a new string and a judged string is significant with respect to the correctness of the answer. Document retrieval test collections do not have this problem because the unique identifiers assigned to the documents makes it trivial to decide whether or not a document retrieved in a new run has been judged.

The problems caused by not having a reusable test collection are illustrated by the plight of the researchers at the University of Ottawa. The pair of runs they submitted to the QA track were misnumbered, and the mistake was not discovered until judging was complete. They were unable to get an official score for their correctly numbered run because they could not map their answer strings to the judged answer strings. As an approximate mapping, they marked any string that completely contained a string that was judged correct in the official adjudicated qrels as being correct, and all other strings were marked as incorrect [10]. This is a conservative mapping that will almost never mark a string correct that would have been judged incorrect at the expense of marking as incorrect many strings that would have been judged correct. As such it provides a lower bound on the score of the unjudged run, but is sufficiently biased against unjudged runs to make the scores between judged and unjudged runs incomparable.

MITRE also developed an approximate mapping technique based on their work with reading comprehension tests [3]. A human created an answer key for the question set, and strings are then marked based on word recall. If an answer string matches a sufficiently high proportion of the words in an answer key entry, the string is marked correct and otherwise it is marked incorrect. Their preliminary analysis demonstrated a high correlation between low word recall scores and judgments of incorrect, and high word recall scores and judgments of correct across the set of runs submitted to the QA track.

We created an approximate mapping algorithm similar in spirit to word recall. An answer key consisting of a set of Perl string-matching patterns was derived (by a human) from the set of strings that were judged correct in the adjudicated qrels. An answer string that matches any pattern for its question is marked correct, and is marked incorrect otherwise. The patterns were created such that almost all strings that were judged correct would be marked correct, sometimes at the expense of marking correct strings that were judged incorrect. For example, patterns for "who" questions generally consisted of the last name of the correct answer. Patterns were constrained to match at word boundaries and case was ignored.

	50-byte Run		250-byte Run	
	MRR	# Not Found	MRR	# Not Found
Human	.291	117	.432	87
Patterns	.292	117	.467	78

Table 3: Mean reciprocal rank (MRR) and number of questions for which no answer found (# Not Found) for human and pattern-based judgments for originally unjudged runs.

A total of 338 patterns was created for the 198 questions in the test set, for an average of 1.71 patterns per question. The majority of questions (128) had a single pattern, and 41 additional questions had just two patterns. At the other end, one question had 13 patterns and another had 10 patterns. The question with 13 patterns was *What does the Peugeot company manufacture?*, which had a variety of accepted answers including cars, automobiles, diesel motors, plastic components, and electric vehicles. In addition, assessors accepted various model numbers such as Peugeot 405s, 309s, 106s, 504s, 505s, 205s, and 306s.

Each of the 41 QA track runs was re-scored using the pattern matching judgments. The Kendall correlation between the system rankings produced by the adjudicated qrels and the pattern-matching judgments was .96, the average level of correlation seen when using different human assessors for judged runs. So, is the problem of creating a reusable QA test collection solved? Not really. Unlike the different judgments among assessors, the approximate mapping techniques misjudge broad classes of responses—classes that are usually precisely the cases that are difficult for the original QA systems. For example, an answer string containing multiple entities of the same semantic category as the answer will always be judged correct if the correct answer is mentioned. Document context is also not taken into account. A more sophisticated pattern matching technique could eliminate some of this problem by conditioning patterns by the document ids that are in the judged set, so, for example, questions such as *Who is the prime minister of Japan?* only accept certain names for certain time ranges. But this does not solve the problem for completely wrong context that happens to contain a correct string.

Another concern with the computed correlation between the pattern-based judgments and the adjudicated qrels is the fact that the patterns were created from the responses of the systems that were then re-scored; this is essentially testing on the training data. The two University of Ottawa runs provide an opportunity to test the pattern matching scoring technique on runs that did not contribute to the judgments from which the patterns were drawn. For this test, the two Ottawa runs were judged by a human[3] and also scored by the patterns. The mean reciprocal rank and number of questions for which no answer was found according to both judgment sets are given in Table 3.

The rank of the first response to be marked correct for the 50-byte run differed between the two judgments sets for 11 of the 198 questions. The human judgments gave a higher score for 5 of those questions. For the 250-byte run, 14 questions differed, and the human judgments gave a better score for 2 of those questions. In all of the cases in which the human judgment gave a better score, the pattern would have been created to accept the string had the string been in the original judgment set. Seven of the 18 cases in which the patterns gave a higher score involved responses that had multiple entities with the same semantic category as the answer but did not indicate which was the answer. Other cases included such problems as accepting "Giacomo Joyce" when the answer was James Joyce (pattern is "Joyce"); accepting Plainfield, N.H. as the location of Dartmouth College (which is in Hanover, N.H.) because the pattern accepts "N.H."; and accepting a quote from a spokesman at the "India Embassy" for the location of the Taj Mahal (pattern is "India").

These examples suggest that the concerns about pattern-based judgments are well-founded. Clearly the patterns (or similar methods such as word recall) are better than nothing, especially for experiments internal to an organization where the benefits of a reusable test collection are most significant (and the limitations are likely to be well understood). However, the full benefits of test collection evaluation will not be fully realized for the QA task until more satisfactory techniques for evaluating new runs are devised.

5 Conclusion

The primary purpose of the TREC-8 Question Answering Track was to promote research on the question answering task. A secondary goal was to provide a venue for investigating the appropriateness of using an evaluation methodology based on assessor judgments for a task other than document retrieval.

As the first large-scale evaluation of domain-independent question answering systems, the results of the track document the current state-of-the-art. For this constrained version of the question answering task, the most accurate systems found a correct response for more than 2/3 of the questions. When an answer was found at all, it was usually highly ranked. Relatively simple bag-of-words approaches were adequate for finding answers when responses could be as long as 250 bytes (roughly equivalent to a sentence or two), but more sophisticated approaches were necessary for responses of less than 50 bytes. In general, these approaches involved classifying a question by the type of entity that would answer it, and using shallow parsing to find entities of the appropriate type close to the question's words in a document.

Analysis of the evaluation methodology used in the track demonstrated that it was both appropriate and effective, although it did not result in a reusable test collection. Assessors had differences of opinion as to whether a supplied answer string actually answered the question. They differed on how much of a name is required, and on the granularity of times and locations. If assessors have different opinions as to what constitutes an answer, then eventual end-users of the technology will have different opinions as well; the technology must accommodate user differences to be useful. Fortunately, systems can be meaningfully compared even though judgment sets produced by different assessors differ. The different judgment sets do produce different absolute scores for the same run, but the relative scores across runs are quite stable.

When assessor judgments are based on the entire answer string, the resulting judgment set does not create a reusable test collection. The problem is that two runs seldom return exactly the same string as a response to a question, and there is currently no completely accurate way to to map specific answer strings to the more general answers. Furthermore, the linguistic processing required of such a mapping is equivalent to that needed to solve the original QA problem. This is in contrast to document retrieval for which it is trivial to decide whether a judged document has been retrieved by a different run. Several approximate mappings have been suggested for the QA problem, including

[3]The original NIST assessors were not available to judge these runs, but the adjudicator for the official qrels did the judging. Thus, the judgments followed the same rules as the official judgments.

the approach based on pattern matching introduced in this paper. When tested on the track results, the ranking of systems produced by pattern-based judgments differed from the ranking based on the adjudicated judgments by an amount equivalent to that of using different human assessors. However, the patterns misjudge entire classes of response strings, including many of the more interesting cases, and the issue of evaluation bias for unjudged runs has just begun to be explored. Reusable test collections—which allow researchers to run their own experiments and receive rapid feedback as to the quality of alternative methods—are key to advancing the state-of-the-art. The benefits of TREC-style evaluation will not be fully realized for the QA task until there are adequate techniques for evaluating new runs from the results of judged runs.

References

[1] D.E. Appelt, J.R. Hobbs, J. Bear, D. Israel, M. Kameyama, A. Kehler, D. Martin, K. Myers, and M. Tyson. SRI International FASTUS system MUC-6 test results and analysis. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 237–248. Morgan Kaufmann, 1995.

[2] BBN Systems and Technologies. BBN: Description of the PLUM system as used for MUC-6. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 55–69. Morgan Kaufmann, 1995.

[3] Eric Breck, John Burger, Lisa Ferro, David House, Marc Light, and Inderjeet Mani. A sys called Qanda. In *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*, pages 443–451, November 1999. Notebook draft.

[4] Robin D. Burke, Kristian J. Hammond, Vladimir A. Kulyukin, Steven L. Lytinen, Noriko Tomuro, and Scott Schoenberg. Questions answering from frequently-asked question files: Experiences with the FAQ Finder system. Technical Report TR-97-05, The University of Chicago, Computer Science Department, June 1997.

[5] Paul Cohen, Robert Schrag, Eric Jones, Adam Pease, Albert Lin, Barbara Starr, David Gunning, and Murray Burke. The DARPA high-performance knowledge bases project. *AI Magazine*, pages 25–49, Winter 1998.

[6] Laura L. Downey and Dawn M. Tice. A usability case study using TREC and ZPRISE. *Information Processing and Management*, 35(5):589–603, 1999.

[7] Boris Katz. From sentence processing to information access on the world wide web. Paper presented at the AAAI Spring Symposium on Natural Language Processing for the World Wide Web, 1997. Electronic version at `http://www.ai.mit.edu/people/boris/webaccess`.

[8] Julian Kupiec. MURAX: A robust linguistic approach for question answering using an on-line encyclopedia. In Robert Korfage, Edie Rasmussen, and Peter Willett, editors, *Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 181–190, 1993. Special issue of the SIGIR FORUM.

[9] M.E. Lesk and G. Salton. Relevance assessments and retrieval system evaluation. *Information Storage and Retrieval*, 4:343–359, 1969.

[10] Joel Martin and Chris Lankester. Ask Me Tomorrow: The University of Ottawa question answering system. In *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*, pages 575–583, November 1999. Notebook draft.

[11] John O'Connor. Answer-passage retrieval by text searching. *Journal of the American Society for Information Science*, pages 227–239, July 1980.

[12] Linda Schamber. Relevance and information behavior. *Annual Review of Information Science and Technology*, 29:3–48, 1994.

[13] Alan Stuart. Kendall's tau. In Samuel Kotz and Norman L. Johnson, editors, *Encyclopedia of Statistical Sciences*, volume 4, pages 367–369. John Wiley & Sons, 1983.

[14] Ellen M. Voorhees. Variations in relevance judgments and the measurement of retrieval effectiveness. In W. Bruce Croft, Alistair Moffat, C.J. van Rijsbergen, Ross Wilkinson, and Justin Zobel, editors, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 315–323, Melbourne, Australia, August 1998. ACM Press, New York.

[15] Ellen M. Voorhees. Special issue: The sixth Text REtrieval Conference (TREC-6). *Information Processing and Management*, 36(1), January 2000.

[16] Ellen M. Voorhees and Dawn M. Tice. The TREC-8 question answering track evaluation. In E.M. Voorhees and D.K. Harman, editors, *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*. Electronic version available at http://trec.nist.gov/pubs.html, 2000.

[17] B. Webber. Question answering. In Stuart C. Shapiro, editor, *Encyclopedia of Articficial Intelligence*, volume 2, pages 814–822. Wiley, 1987.

[18] Terry Winograd. Five lectures on artificial intelligence. In A. Zampolli, editor, *Lingusitic Structures Processing*, volume 5 of *Fundamental Studies in Computer Science*, pages 399–520. North Holland, 1977.

[19] W. A. Woods. Lunar rocks in natural english: Explorations in natural language question answering. In A. Zampolli, editor, *Lingusitic Structures Processing*, volume 5 of *Fundamental Studies in Computer Science*, pages 521–569. North Holland, 1977.