Comparing an Integrated QA system performance - A Preliminary Model

Samira Shaikh[1], Tomek Strzalkowski[1,2], Sarah M. Taylor[3],
Jonathan L. Smith[3]

[1]ILS Institute, State University of New York , University at Albany,
New York

[2]Institute of Computer Science, Polish Academy of Sciences

[3]Advanced Technology Office, Lockheed Martin IS&GS

samirashaikh@gmail.com, tomek@albany.edu,
sarah.m.taylor@lmco.com, jonathan.m.smith@lmco.com

Abstract

A complex NLP application such as a question answering system requires sophisticated methods for evaluation, especially if there is a mechanism to submit factoid type queries as well as natural language (NL)-type questions. As part of our research we created an integrated tool that allows users to search two ways – using a keyword query and using a NL question. We present a method to evaluate output from the two types of searches and we explore the correlation of scores when judgments of output are made against keywords and those made against NL questions. We present a preliminary model, which may serve as a guideline to evaluate similar systems' performance.

1 Introduction

Evaluation techniques of QA systems can be *information-based* such as in TREC (Voorhees et al. 2000, Voorhees and Harman, 2005) focusing on objective metrics; or *utility based*, which focus on the usability of the QA system for the user and other subjective measurements. Several other methodologies have been proposed for QA applications such as automatic evaluation (Breck et al. 2000) and evaluation on multiple dimensions (Nyberg et al. 2002). While these paradigms address one or more aspects of evaluation there have not been many that explore the correlation between judgments and the source query against which the judgments are made. For instance, when a system's response is judged against a full NL question, how the resulting score compares to when the same response is judged against a keyword query. This is of course a perennial problem in information retrieval where user queries are routinely truncated to indexing keywords; however, in QA applications where the expectations of answer precision are much higher than in document retrieval, such considerations are harder to ignore.

Our evaluation method is user-centered, in that scores are provided by a set of judges, by classifying system output to sets of predetermined relevance groups. This methodology has been explored in detail, (Kelly et al. 2006, Lin 2008, Roberts et al. 2004), however, attempts to combine evaluation the two different question types along with two different retrieval methods have not been studied in depth.

We utilize judging techniques outlined by Small et al. (Small 2006) as a guideline for performing component evaluation. In addition to an end-to-end QA evaluation, our focus in part of the research was the information extraction (IE) component of the system. IE evaluation methodology has been greatly developed at the MUC conferences (Hirschman 1998), but our focus here is to attempt to combine the evaluation of information extraction with that of QA and endeavor to find correlation between the two. We wish to ascertain whether the effect of performance of a component in a complex QA system is transitively reflected upon final system outcome.

Section 2 of this paper describes the experimental setup behind the evaluation and Section 3 details the findings and results of the experiments.

2 Experimental Setup

We integrated two types of tools as part of our research. Timeline (Taylor et al. 2006) is a tool that searches structured database of events using Boolean queries. These events are extracted from text sources and stored in a database along with the corresponding text passages. Users may input complex Boolean queries consisting of sophisticated keywords and Boolean operators (AND, OR). The system then retrieves matching events and displays them as icons on a graphical timeline, while the original passages are shown to the user (when the mouse is moved over corresponding icons).

COLLANE (Strzalkowski et al. 2007; 2009) is the other part of the integrated system. It allows the users to input natural language questions and search the original text sources, including newly arriving data. COLLANE uses complex NL techniques such as parsing of sentences and user questions, framing events found in text and robust interactive dialogue to answer user questions. It also includes the AeroText information extraction module (Taylor, 2004), which extracts events from text, in addition to named entities.

Thus the integrated system has the capability of searching its database via the structured Boolean queries (by matching event templates) as wells as via complex NL questions (by passage-based QA). We are interested which of the access methods may be more effective from the user viewpoint.

System integration was achieved using service oriented architecture and the Web 2.0 paradigm. We created a COLLANE web service, which takes as input an NL question and returns a list of relevant passages to the calling service. Timeline users have the option to choose which type of query they wish to submit. A keyword query is matched against Timeline's event database, events are retrieved and the passages that these events are extracted from make the output of the system. If the user input is an NL question, it is sent to COLLANE's QA web service via the web; and a ranked list of passages is sent back to Timeline to be displayed as output. Timeline displays the events that correspond to these passages, to make the output of both keyword search and QA search consistent.

These events, both for the database search in Timeline or for QA using COLLANE serve as a pointer to salient information in the passages. Users can create complex timeline structures using these events and organize their arguments effectively. Events extracted by AeroText information extraction module are structured templates with multiple entities and their roles.

3 Evaluation

We first ran a preliminary evaluation to determine the efficacy of the adopted paradigm. For this experiment we assumed that the NL question and keyword query are equivalent; in that the keyword queries are 'derived' from the natural language questions.

3.1 Test Set

A test data set was created by combining a set of 398 documents that were mined off the web in September 2008 and a set of 2690 documents retrieved from the National Counter Terrorism Center web blog. The blog articles were dated from the years 2005 to 2006. This created a set of approximately 100 MB of textual data mostly containing articles in the counter-terrorism domain. We generated a set of 25 test questions and queries pertaining to the dataset. To consistently judge the performance of the keyword search module against the NL module of this integrated system, and *vice versa*, the keyword search queries were 'derived' from the NL questions; we removed stop words and stemmed compound words to generate a keyword query from the question.

For example, if the natural language question is:

'*What roadside and IED bombings were there in Iraq in 2005?*'

then the corresponding keyword query becomes:

'*roadside IED bombing Iraq 2005*'

3.2 Initial Run

	Database Search	NL query search
Total queries run	25	25
Events Retrieved	16512	618
Avg.	660.48	24.72

Table 1. Events Retrieved

Table 1 shows the results of running the 25 questions and the corresponding queries through the two tools. Clearly, the automatically derived Boolean queries are very inaccurate since the keywords are by default interpreted as with OR conjunction. Thus, many unwanted matches are returned. Obviously, most users would not normally settle for such a query; however, many users may well start with it.

3.3 Second Experiment

We used the same data set for testing and the same NL question set as was used in the preliminary run. However, this time we obtained a set of Boolean keyword queries that an expert user of Timeline would likely pose, rather than using a simple 'OR' query of all the major keywords in the NL question. It turned out that the expert user typically start with AND connected queries and then gradually relax them by dropping or replacing terms, until they feel the results are satisfactory.

For example, if the natural language question is:

'*What suicide bombings were there in Iraq in 2005?*'

then the corresponding keyword 'OR' query became:

'*suicide bombings Iraq 2005*'

which is equivalent to:

suicide OR bombings OR Iraq OR 2005

while the expert query for the same would be:

suicide AND bomb AND Iraq AND 2005

	Database Search	NL query search
Total queries run	25	25
Events Retrieved	57	618
Avg.	2.47	24.72

Table 2. Events Retrieved using expert queries

As expected (cf. Table 2), there is a dramatic decrease in the number of events retrieved by Timeline database search when using queries that are not a simple disjunction of terms. On the flip side, using AND connected terms leads to over-specification and as a result few relevant events are found. As a consequence, expert searchers tend to eliminate certain keywords and repeat searches until some results are returned although this may result in low search precision, as we will see below.

The results in Tables 1 and 2 also demonstrate that going from natural language questions, which are formed intuitively in a user's mind to a strict Boolean query may result in a drop in both precision and recall. Some salient elements of the information need are either lost or left underrepresented. These results are consistent with other system performance evaluations that investigate Boolean retrieval strategies. (Saggion 2004)

3.4 NL Question Search Evaluation

We asked a set of judges to rate the results of the run from the integrated Timeline-COLLANE system. Since there are two sets of output for each type of search – a set of passages and a set of events extracted from these passages – we tasked the judges with rating the relevance of both these sets. In the first step of this process, judges were presented with the natural language questions as posed to COLLANE and a series of passages returned. They were asked to judge each of the passages into one of the 3 categories: relevant (if the passage contains the information asked for in the question), non-relevant (if the passage was not related to the question or conflicted with some of the request parameters, e.g., dates); or "cannot judge" (when the passage did not contain sufficient information to decide relevance). We note that items in this third category may in fact point to relevant information, but judging this would require reviewing the source document. In real life applications, end user may follow such underspecified leads depending upon some external factors such as available time and criticality of the topic.

	% Relevant	% Non-Relevant	% Cannot Judge
Judge 1	45.74	53.83	0.43
Judge 2	19.36	74.68	5.97
Avg.	32.55	64.25	3.20

Table 3. Passage relevance judgments judged against NL questions

Table 3 shows the relevance judgments for the passages that were retrieved by COLLANE for the test question set. We note that there is a fairly significant level of disagreement between the judges, which again has to do with the limited context in which these judgments are made. In the second step of the process, two different judges were presented with the same questions, but this time only the events extracted from the passages were displayed for judgment, not the text passages from which they came. In other words, the context in which to arrive at the judgment was further limited.

	% Relevant	% Non-Relevant	% Cannot Judge
Judge 1	2.26	60.52	37.22
Judge 2	1.04	68.00	30.96
Avg.	1.66	64.26	34.08

Table 4. Event relevance judgments judged against NL questions

As expected, the relevance scores for events were quite low (Table 4); however, the consistency between the judges did improve as seen in their rates of non-relevant scored events. While the fewer positive judgments were clearly due to incomplete information contained in the event representation, the negative (non-relevant) judgments were now easier to make (e.g., when event attributes such as *time* were inconsistent with the question). For the same reason, the percentage of events that were classified as ‘Cannot judge’ was significantly greater than for passages; nonetheless, we expect that the majority of “cannot judge” events would in fact pointing to relevant information. This is in line with the passage relevancy scores, but also reveals that judgments based solely on computer-generated structures are often difficult to make.

An alternative way to evaluate event extraction is to take into consideration the ratings of corresponding passages. In order to avoid judge bias, we have devised the following strategy that makes event scores more consistent with the scores of the underlying passages:

1. If the passage is scored RELEVANT and the event extracted from it is scored either CANNOT-JUDGE or RELEVANT then the event is scored as RELEVANT
2. If the event is scored NON-RELEVANT then it remains NON-RELEVANT, irrespective of the passage score.
3. If the passage is scored NON-RELEVANT then all events extracted from this passage are also scored NON-RELEVANT.

Figure 1. Strategy to judge event relevance based on passage relevance

Condition 1 corresponds to the common case when correct but incomplete information is extracted from a passage into the event structure. Such incomplete references may be considered as positives as long as they point to information that a user can judge as relevant. The second condition corresponds to the situation when the event representation extracted from a passage text contains information that makes the event non-relevant to the question; for example, if the dates are incompatible with the request. This often signifies than the passage itself is not relevant, but it may also arise in situations when the system made a mistake when constructing event attributes. We decided that such misleading references couldn’t be judged positively. The last condition guards against assigning positive scores to spurious “events” derived from non-relevant passages.

Using the above strategy we recalculated the average scores for events as shown in Table 5 below. It should be noted that while these results are more in line with the passage scores, their practical significance might be limited.

	% Relevant	% Irrelevant
Revised Avg. Score	40.49	59.51

Table 5. Event relevance judgments adjusted by passage relevance

3.5 Database Query Search Evaluation

Next, we asked our judges to rate the results of running the Boolean keyword queries against the structured database. The judges were presented with Boolean keyword queries (usually two, three or four words) and a series of event structures returned from database search (Table 7). While the queries are derived from NL questions, the judges could view only the final expert-made queries and judge the results based on these. We also had the judges rate the passages retrieved for the database search (Table 6). In order to obtain these judgments we used the scoring strategy outlined in Figure 1, thus eliminating the need for the Cannot-judge category.

As with the natural language question search reported before, passages received a greater percentage of relevance judgments than events. Tables 6 and 7 show higher precision rates than those reported in Tables 3 and 4; however, these numbers are not directly comparable because the latter assessments were made based on viewing the full NL questions rather than 1-3 isolated keywords. Accordingly, we asked the judges to evaluate the same set of results from the Timeline Database search but this time basing their assessment on the full NL questions rather than just the keywords used in search (Tables 8 and 9).

Also, output judged against keyword query shows higher percentages of relevance than when judged against natural language questions. This is true for both, passage and events. We postulate that this is due to the inherent ambiguity in a keyword query. We noted instances in the judgments where the same passage was scored as relevant to the keyword query while it is was scored as irrelevant to the natural language question. This may point towards the occurrence of false positives in the relevant set for keyword judgments.

Another trend that we observe is that passages have higher relevance score higher than events and that when events scores are higher, so are passage scores. Table 7 shows higher event relevance scores and correspondingly passage relevance scores are higher in Table 6, as compared to Tables 8 and 9.

	% Relevant	% Irrelevant
Judge 1	82.46	17.54
Judge 2	77.19	22.81
Avg.	79.82	20.18

Table 6. Passage relevance judgments against keyword query

	% Relevant	% Irrelevant
Judge 1	12.20	87.80
Judge 2	70.73	29.27
Avg.	41.46	58.54

Table 7. Event relevance judgments against keyword query

	% Relevant	% Irrelevant
Judge 1	52.63	47.37
Judge 2	54.39	45.61
Avg.	47.37	46.49

Table 8. Passage relevance judgments judged against NL questions

	% Relevant	% Irrelevant
Judge 1	31.58	68.42
Judge 2	39.16	60.84
Avg.	35.37	64.63

Table 9. Event relevance judgments judged against NL questions

The results in Table 8 now show a better correlation with those in Table 3 when the NL queries are run through the integrated system. The direct comparison is still not possible due to differences in recall levels at which precision is measured. Since natural language search in this integrated system returns approx. 25 passages per query at about 35% precision, we obtain about 8 relevant passages per question. On the other hand, Boolean keyword search returns only about 2 passages per query at 50% precision, which gives us only 1 relevant passage per query. In order to obtain a more meaningful comparison we need to obtain a larger pool of judgments to allow for an estimation of recall.

4 Conclusions

We compare output from two different types of searches using two different question formats. We found that system output judged against Boolean keywords queries is scored higher than when judged against natural language questions. However, the occurrence of false positives in the relevant set for keywords queries needs to be explored further. We also investigated the effect of component output on final system output. We note that the number of judges is limited for this evidence to be conclusive. The evaluation results are motivating and need to be explored in depth with a larger data corpus and more judges. This preliminary model may be extended further to evaluate system output where two complex end-to-end systems are integrated together, a trend which is prevalent in these days of web services and loosely coupled integration.

References

Breck, Eric J. and Burger, John D. and Ferro, Lisa and Hirschman, Lynette and House, David and Light, Marc and Mani, Inderjeet. How to Evaluate your Question Answering System Every Day and Still Get Real Work Done. In *Proceedings of the Second Conference on Language Resources and Evaluation (LREC-2000).*

Hirschman, L. 1998. The evolution of evaluation: Lessons from the message understanding conferences. *Computer Speech and Language* 12:281–305.

Horacio Saggion, Rob Gaizauskas, Mark Hepple, Ian Roberts and Mark A. Greenwood. Exploring the Performance of Boolean Retrieval Strategies For Open Domain Question Answering. *IR4QA: Information Retrieval for Question Answering, SIGIR 2004*

I. Roberts and R. Gaizauskas. Evaluating passage retrieval approaches for question answering. *In Advances in Information Retrieval: Proceedings of the 26th European Conference on Information Retrieval (ECIR04), number 2997 in LNCS, pages 72{84, Sunderland, 2004. Springer*

Kelly D. and Kantor P. B. Questionnaires for Eliciting Evaluation Data from Users of Interactive Question Answering Systems. *In the Special Issue of The Journal of Natural Language Engineering. HLT 2006*

Jimmy Lin and Mark D. Smucker. How Do Users Find Things with PubMed? Towards Automatic Utility Evaluation with User Simulations. *Proceedings of the 31th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2008)*, pages 19-26, July 2008, Singapore.

Nyberg, E and Mitamura, Teruko. 2002. Evaluating QA Systems on Multiple Dimensions. *In Proceedings of the Workshop on QA Strategy and Resources*

Small, S. and Strzalkowski, T. HITIQA: High-Quality Intelligence through Interactive Question Answering. *In the Special Issue of The Journal of Natural Language Engineering. HLT 2006*

Strzalkowski, T. , S. Small, S. Shaikh, S. Taylor, B. Lipetz, H. Hardy, T. Cresswell. 2007. Collaborative Analytical Workshop with COLLANE, Preliminary Report. *IARPA CASE Program.*

Strzalkowski Tomek., Sarah Taylor, Samira Shaikh, Ben-Ami Lipetz, Hilda Hardy, Nick Webb, Tony Cresswell, Min Wu, Yu Zhan, Ting Liu, and Song Chen (forthcoming, 2009). COLLANE: An experiment in computer-mediated tacit collaboration. In *Aspects of Natural Language Processing* (M. Marciniak and A. Mykowiecka, editors). Springer.

Taylor, Sarah M. (2004). "Information Extraction Tools: Deciphering Human Language." IT Professional. Vol. 06, no. 6, pages: 28-34. November/December, 2004. Online. http://ieeexplore.ieee.org/iel5/6294/30282/01390870.pdf?tp=&arnumber=1390870&isnumber=30282

Taylor, Sarah M., Cassel, David., Katz, Gary., Childs, Lois., Rimey, Raymond. *In Proceedings of Intelligent Tools Workshop.* 2006.

Voorhees, E. M. and Harman, D. K. (2005) TREC: Experiment and Evaluation in Information Retrieva. *Cambridge, MA: MIT Press.*

Vorhees. E and Tice. D. 2000. Implementing a Question Answering Evaluation. *In Proceedings of LREC'2000 Workshop on Using Evaluation within HLT Programs: Results and Trends*