# Overview of the TREC 2006 ciQA Task

Diane Kelly[1] and Jimmy Lin[2]

[1]School of Information and Library Science
University of North Carolina at Chapel Hill
Chapel Hill, NC 27599
dianek@email.unc.edu

[2]College of Information Studies
University of Maryland
College Park, Maryland, USA
jimmylin@umd.edu

**Abstract**

Growing interest in interactive systems for answering complex questions lead to the development of the complex, interactive QA (ciQA) task, introduced for the first time at TREC 2006. This paper describes the rationale and design of the ciQA task and the evaluation results. Thirty complex relationship questions based on five question templates were investigated using the AQUAINT collection of newswire text. Interaction forms were the primary vehicle for defining and capturing user-system interactions. In total, six groups participated in the ciQA task and contributed ten different sets of interaction forms. There were two main findings: baseline IR techniques are competitive for complex QA and interaction, at least as defined and implemented in this evaluation, did not appear to improve performance by much.

## 1 Introduction

Recent publications, interests of funding sponsors, and discussions at TREC and other conferences, indicate that the field of question answering is moving in two major directions:

- A shift away from "factoid" questions towards more complex information needs that exist within richer user contexts.
- A shift away from purely systems-oriented evaluation towards methodologies based at least in part on interactions with users.

In short, there appears to be growing interest in interactive systems designed to handle complex questions. The ciQA task, which reflects a departure from previous TREC QA evaluations, was devised to reflect these trends. The ciQA task combined elements from the TREC 2005 relationship task (Voorhees and Dang, 2005) and the TREC 2005 HARD track (Allan, 2005), which focused on single-iteration clarification dialogues.

# 2 Evaluation Design

The ciQA evaluation differed from other QA evaluations in that it followed a multi-step procedure: In the first round, participants submitted initial runs and "interaction forms". NIST assessors then interacted with these forms—the results of which were returned to participants. In the second round, participants submitted final runs based on the feedback solicited through the interaction forms. NIST then evaluated both the initial and final runs. By comparing the two runs, it becomes possible to quantify the effects of the interactions. Two key components of the ciQA evaluation are discussed below—interaction and complex questions.

## 2.1 Interactive QA

The decision about how to implement the interactive component of the evaluation was arrived at after careful consideration of many issues, including lessons from two previous TREC tracks—interactive and HARD. The TREC interactive track (Hersh and Over, 2001; Dumais and Belkin, 2005) lasted nine years and attempted to integrate user studies into a batch-style evaluation framework. Although the track experimented with different approaches, it ultimately converged on a framework where some elements of the study design (such as the corpus, topics, instruments, measures, and protocols) were standardized, while other aspects of the study design (such as retrieval systems, interfaces, and research questions) varied across sites. Participants were responsible for recruiting and running a specified number of subjects. The design meant that subjects were only able to complete a small number of tasks during search sessions (approximately 4–6) since searches typically lasted around 15 minutes per topic. The number of subjects and topics, and the duration of user–system interactions represented a delicate balance—influenced to a large extent by the physical and cognitive limitations of human searchers. While the track made many contributions to the evaluation of interactive information systems, the evaluation model was limited. Experiments were not directly comparable across different sites and the findings were often hard to generalize because of the impact of the searchers, systems, and topics.

The TREC HARD track took a different approach to studying the potential benefits of user–system interactions (Allan, 2005). It introduced the idea of clarification forms, from which ciQA interaction forms are derived. The clarification form provided track participants with an opportunity to elicit feedback from assessors. Typical approaches included various implementations of term relevance feedback, where terms, phrases, or passages were presented to assessors who then evaluated the items for relevance. This information was then used by the system for such techniques as query expansion.

One important difference between the interactive and HARD tracks was that in the latter the nature of the interaction was held constant to a certain extent since users interacted in a prescribed format. Another important difference was the nature of the "user"—in the interactive

**Template:** What evidence is there for transport of [drugs] from [Mexico] to [the U.S.]?
**Narrative:** The analyst would like to know of efforts to curtail the transport of drugs from Mexico to the U.S. Specifically, the analyst would like to know of the success of the efforts by local or international authorities.

Figure 1: A complete sample ciQA topic.

track, participants recruited users from their sites, while in the HARD track the user *was* the NIST assessor. This distinction is important since assessors are also topic creators. Assessors for HARD worked on their own information needs, rather than someone else's. Thus, in the HARD track users were also held constant to a certain extent. By standardizing the interaction format and leveraging NIST assessors as the common "user", it was possible to develop shared tasks that had the potential to yield comparable results. To accomplish this with limited resources, however, it was necessary to simplify and abstract user–system interactions.

The ciQA task followed in this philosophy of trading a degree of insightfulness for affordability and repeatability. Limiting the scope and duration of the interactions made large-scale evaluation more practical and cost effective. There are other reasons to study the single-iteration feedback cycle: users in the Web-centered information age have become acclimated to a rapid back-and-forth style of interaction. Any interactive system that doesn't produce better results with limited user input is unlikely to gain widespread adoption. We believe that a single unit of interaction is a reasonable starting point in the exploration of interactive QA. Thus, for the purposes of the ciQA evaluation, we considered the smallest possible interaction unit to consist of a user responding to a system and the system using the user's responses to produce new content—this corresponds to what Spink calls an interactive feedback unit (Spink, 1997). The ciQA task captured one iteration of this cycle.

## 2.2 Complex "Relationship" Questions

The second component of the ciQA evaluation was the use of complex relationship questions. The ciQA task extended and refined the so-called "relationship" task piloted in TREC 2005. A relationship was defined as the ability of one entity to influence another, including both the means to influence and the motivation for doing so. Eight "spheres of influence" were noted in a previous pilot study funded by AQUAINT: financial, movement of goods, family ties, communication pathways, organizational ties, co-location, common interests, and temporal. Evidence for both the existence or absence of ties is relevant. The particular relationships of interest naturally depend on the context.

There are two reasons for this choice of task. Questions of this sort have be identified as a class of information needs commonly faced by intelligence analysts today. Building on a previous task also provides continuity and training data for participants.

A relationship question in the ciQA task, which we refer to as a topic (to reduce confusion and to better align with established TREC parlance), is composed of two parts—a complete example is shown in Figure 1. The question template is a stylized information need that has a fixed structure and free slots (the bracketed items) whose instantiation varies across different

What evidence is there for transport of [goods] from [entity] to [entity]?
**Example:** What evidence is there for transport of [drugs] from [Mexico] to [the U.S.]?

What [relationship] exist between [entity] and [entity]?
(where [relationship] is an element of {"financial relationships", "organizational ties", "familial ties", "common interests"})
**Example:** What [financial relationships] exist between [drug companies] and [universities]?

What influence/effect do(es) [entity] have on/in [entity]?
**Example:** What effect does [aspirin] have on [coronary heart disease]?

What is the position of [entity] with respect to [issue]?
**Example:** What is the position of [John McCain] with respect to [the Moral Majority or the Christian Coalition]?

Is there evidence to support the involvement of [entity] in [event/entity]?
**Example:** Is there evidence to support the involvement of [China] in [human organ transplants from Chinese prisoners]?

Figure 2: The five template types developed for the ciQA task, with an example instantiation each.

topics. The narrative is free-from natural language text that elaborates on the information need, providing, for example, user context, a more articulated statement of interest, focus on particular topical aspects, etc. In total, five template types were developed for the ciQA task. They are shown in Figure 2, with example instantiations.

Why templates? Task analysis reveals that users' questions can often be categorized into generic prototypes. In the TREC genomics track, for example, templates are used to capture stylized questions in the biomedical domain (Hersh et al., 2005). DARPA's GALE program similarly uses question templates to define its "distillation" task. The use of templates in ciQA reflects this emerging trend—but since they are not a "one-size-fits-all" solution, topics are augmented with free-text narratives.

In addition, the ciQA topic structure also lowers the barrier of entry for non-QA researchers who might be interested in the retrieval aspects of the task. Currently, a successful QA system requires the integration of many components (IR, linguistic analysis, named-entity recognition, etc.)—this complexity serves as a serious hurdle for new researchers making forays into the field. Question templates are pre-structured, which potentially simplifies the analysis that a system must perform—thus helping to "level" the playing field.

# 3 Evaluation Procedure

The ciQA evaluation was implemented in TREC 2006 as a secondary task in the QA track (the main task being focused on question series). In total, thirty topics were developed (six for each template). The corpus employed in the experiments was the AQUAINT collection of newswire text,[1] which is comprised of approximately one million documents totaling roughly three gigabytes.

To support the individual goals of participants, the interactive aspect of ciQA was optional. Those wishing to explore interactivity submitted interaction forms to NIST along with their initial runs. Both automatic and manual runs were allowed. A manual run was defined as any submission with human intervention, i.e., in the preparation of runs, generation of interaction forms, and exploitation of user feedback.

In ciQA, user–system interactions were encapsulate in HTML pages (i.e., interaction forms) and interaction types were limited to elements that could appear on an HTML form—checkboxes, radio buttons, text input boxes, and the like. The results of the interactions were restricted to that which could be captured through the CGI protocol. However, the availability of Web-based programming languages meant that interaction forms could have theoretically encoded arbitrarily complex system behaviors, since Javascript was allowed. Each topic was associated with a unique form, whose content was up to the discretion of the participant. Participants were allowed to submit two different sets of clarification forms, which provided an opportunity for exploring and comparing alternative forms.

NIST assessors spent no more than three minutes per topic on each form. This included the time needed to load the form, initialize any content, and render it. The presentation order of forms was rotated across topic. At the end of three minutes, if the assessor had not submitted the form, it timed out and was forcibly submitted. CGI variable bindings associated with the forms captured the result of the interactions, which NIST returned to the participants approximately two weeks after the original submission (along with timing statistics).

System answers were limited to 7,000 non-whitespace characters in length and were assessed with the nugget-based methodology used in previous TREC evaluations of complex questions (Voorhees, 2003). This method is based on assessors identifying "information nuggets" contained in system responses and captured using an F-score that places heavy emphasis on nugget recall ($\beta = 3$). In addition, nugget pyramids (Lin and Demner-Fushman, 2006) were implemented to obtain a more refined notion of nugget importance. Finally, the task also experimented with a novel method for measuring performance in terms of plots that quantify recall against answer length. Note that the assessment process did not begin until the final runs had been received. Initial and final runs were evaluated side by side, but NIST assessors were not made aware of this. For more details, see (Dang et al., 2006).

# 4 Results

The ciQA task in TREC 2006 drew participation from six groups (four from within the United States, and two from outside). NIST received ten baseline runs and eleven final runs. A total of ten sets of different interaction forms were submitted by the six groups.

In addition to the runs submitted by the participants, the University of Maryland prepared as a reference baseline a submission that used simple sentence retrieval techniques. For each topic, the verbatim question template was used as a query to Lucene, which returned the top 20 documents. These documents were then tokenized into individual sentences. Sentences that contained at least one non-stopword from the question were retained and returned as the baseline run (up to the 7,000

[1] LDC catalog number LDC2002T31

|  |  | **Run tags** |  | **Pyramid F-Score** |  |
|---|---|---|---|---|---|
| **Organization** | **Type** | **Initial** | **Final** | **Initial** | **Final** |
| CL Research | automatic | clr06ci1 | clr06ci1r | 0.151 | 0.184 |
| CL Research | automatic | clr06ci2 | clr06ci2r | 0.175 | 0.209 |
| MIT | automatic | csail1 | csailif1 | 0.203 | 0.209 |
| MIT | automatic | csail1 | csailif2 | 0.203 | 0.203 |
| U. Maryland | automatic | UMDA1pre | UMDA1post | 0.224 | 0.180 |
| U. Maryland | manual | UMDM1pre | UMDM1post | 0.316 | 0.350 |
| U. Mass. | automatic | UMASSauto2 | UMASSi2 | 0.171 | 0.160 |
| U. Mass. | automatic | UMASSauto1 | UMASSi1 | 0.133 | 0.150 |
| U. Strathclyde | manual | strath1 | strath4 | 0.227 | 0.239 |
| U. Waterloo | automatic | UWATCIQA1 | UWATCIQA3 | 0.247 | 0.247 |
| U. Waterloo | automatic | UWATCIQA1 | UWATCIQA4 | 0.247 | 0.268 |
| U. Maryland | automatic | sentence retrieval reference |  | 0.237 | 0.264 |

Table 1: Results from the TREC 2006 ciQA task—11 run pairs and the sentence retrieval baseline.

character limit). Sentence order within each document and across the ranked list was preserved. The interaction forms associated with this run asked the assessor for relevance judgments on each of the sentences (relevant, not relevant, don't know). The final run was prepared by simply removing those sentences judged not relevant—this had the effect of pulling more sentences from documents lower in the ranked list.

In total, there were eleven different initial–final pairs submitted by participants. The pyramid F-scores of these run pairs are shown in Table 1. The table also shows the performance of the sentence retrieval baseline.

Surprisingly, the sentence retrieval baseline performed exceedingly well. Only two baseline runs received a higher score, one of which was a manual run. The high baseline performance is consistent with findings from previous TREC evaluations (Voorhees, 2003). Figure 3 shows a scatter plot of the baseline and final F-scores for all eleven run pairs. Points below the reference line $y = x$ represent cases in which interaction actually decreased performance—there were two such cases.

Plots of pyramid recall as a function of response length are shown in Figure 4—this was a new type of evaluation metric developed specifically for complex QA and introduced in the TREC 2006 ciQA task. These graphs attempt to quantify how quickly a user is able to acquire relevant nuggets by reading system output (in a linear fashion). Naturally, curves that rise quicker represent "better" systems. In the left graph, the sentence retrieval baseline is compared against the best automatic run. In the right graph, the sentence retrieval baseline is compared against the best manual run. It is interesting to note that for the automatic runs, these recall plots paint a different picture of performance than the pyramid F-scores. Although UWATCIQA4 achieves a higher F-score than the sentence retrieval baseline (final), the recall plots suggest that a user is able to acquire information more quickly with the baseline system. For the manual run, the pyramid F-score suggests an increase in response quality, although the recall plots show little difference between the nugget content of the pre- and post-interaction responses. It appears that these recall plots are a useful tool for fine-grained diagnosis of system performance.

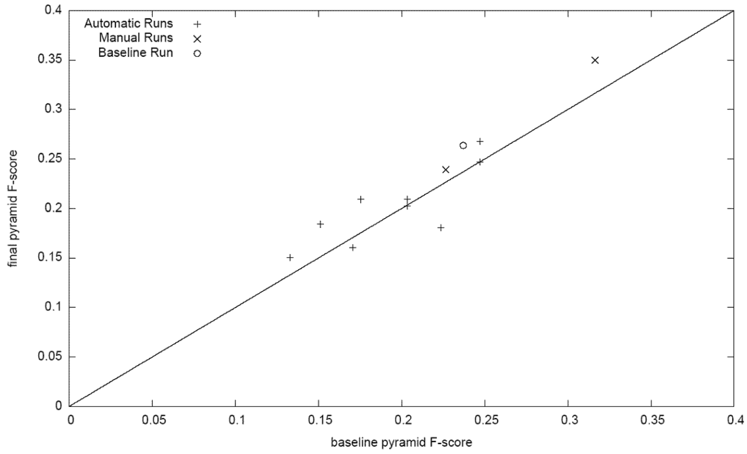

Figure 3: Scatter plot showing baseline and final pyramid F-scores for submitted runs.

# 5 Discussion

Two findings are apparent from the results of the first ciQA evaluation at TREC 2006: that baseline IR techniques are competitive for complex QA, and that interaction doesn't appear to help much (at present).

Most question answering systems today employ a two-stage architecture: IR techniques are first employed to select a candidate set of documents, passages, sentences, etc., which is then analyzed by more sophisticated NLP techniques. For factoid questions, this usually involves named-entity recognition using answer type ontologies. Such techniques provide a lot of value, as they allow systems to pinpoint exact answers. However, does NLP technology help much in complex QA? The sentence retrieval baseline might correspond to the output of a typical first-stage IR module, but ciQA results show that current techniques are not able to improve on this substantially. We believe this is because the factoid QA strategy of anticipating answer types cannot be directly

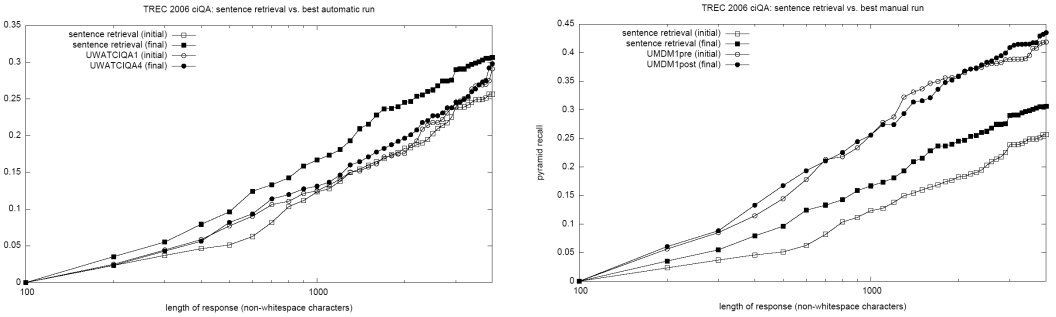

Figure 4: Plots of pyramid recall as a function of response length: sentence retrieval baseline vs. the best automatic run (left) and vs. the best manual run (right)

| Group | Form | Description |
|---|---|---|
| CL Research | 1 | 40 sentences; assessors marked whether each was ”relevant”, ”uncertain”, or ”definitely not”. |
| | 2 | Same as above, except a different technique was used to generate sentences. |
| MIT | 1 | 38 sentences; assessors used check-boxes to indicate sentences that contained answers. Open-ended questions at the end elicited additional keywords or phrases. |
| | 2 | Same 38 sentences as above; assessors used mouse to highlight important text. |
| U. Maryland | 1 | 50 sentences; assessors marked whether each was “relevant”, “somewhat relevant”, or “not relevant”. |
| | 2 | Questions customized to each topic; assessors were also presented with several types of evidence and asked to rank the importance of the evidence. |
| U. Mass. | 1 | Four sections: (1) a list of online newsgroups; (2) a list of terms and alternate spellings; (3) a list of organizations mentioned in results; (4) a list of names mentioned in the results; assessors used check-boxes to select items from each section that were relevant. |
| U. Strathclyde | 1 | 8 possible answers were provided to assessors, who were asked several questions about relevance and quality. At the end of the form, assessors responded to 3 questions about the overall quality of the answer set. |
| | 2 | 6 questions which elicited information about assessors’ context and perceptions of the topic. |
| U. Waterloo | 1 | 15 top-ranking sentences; assessors used check-boxes to indicate sentences that contained answers. |

Table 2: Summary of submitted interaction forms.

applied to complex questions, since their answers do not fall into predictable semantic categories. Since factoid QA techniques are not directly transferable, new approaches need to be developed.

Another observation, evident from Figure 3, is that interaction does not appear to substantially improve QA performance. This finding, we believe, is a conflation of two factors: one concerning interactive QA techniques, the other concerning evaluation design. We discuss these two possibilities below.

Results from the ciQA task may simply reflect the state of the art. Interactive QA, like complex QA, remains in its infancy, especially compared to the more mature task of factoid question answering. Researchers have only begun to explore the space of possible interactions, so it is understandable that large performance gains have yet to be achieved. An analysis of interaction forms is summarized in Table 2. Although participants did not explicitly associate interaction forms with runs, it is reasonable to assume a correspondence between the entries in Tables 2 and 1.

We can see that most participants experimented with variants of relevance feedback—a direct extension of techniques that have worked well in IR tasks. However, there are some important differences to suggest why such approaches may not be entirely effective. First, relevance feedback elicits terms, phrase, documents, etc., that better circumscribe the information need, thereby enhancing recall. In complex QA, however, systems are not rewarded for returning multiple instances of a single nugget. Thus, IR techniques run into the problem of returning "more of the same"—coverage must be balanced with redundancy in a way that is not modeled in document retrieval tasks. Also, complex QA requires system output that is much finer-grained—answers instead of documents. It is unclear if relevance information solicited from users can be exploited at these finer scales.

Although adaptation of IR techniques might be successful, researchers must eventually develop interaction strategies that are unique to question answering and capitalize on its characteristics. The more important question, for the purposes of this work, is whether the ciQA task can serve as a vehicle to guide future progress. Another plausible explanation for the results in Figure 3 is that the evaluation methodology overly restricted the solution space. Perhaps interaction forms are not an effective vehicle for conducting large-scale evaluations of interactive systems. To isolate this possibility, we discuss an alternative implementation of ciQA for TREC 2007 that will help us better understand the issues involved.

# 6 The Future

Overall, we believe that the ciQA task served as an effective tool for formative evaluations of interactive QA systems. Findings should be supplanted by more formal, carefully-designed user studies, but the task provides a method for filtering obviously "bad ideas", thereby helping researchers focus on more promising alternatives.

The encapsulation of user–system interactions into standalone HTML pages submitted to NIST is one possible implementation of interactive QA and represents the approach we have taken in TREC 2006. This embodies a centralized approach that places the assessor–system interactions completely within the control of NIST.

As an alternative, one might consider a distributed evaluation approach, in which participants are requested to host their own systems at a known URL. Rounds of interaction would then proceed by having NIST assessors visit each site. Such a setup would greatly expand the scope of interactions, allowing participants to deploy fully-functional QA systems without being restricted by the interaction forms. However, the unpredictability of network connections, especially to multiple sites around the world, presents serious challenges to smooth coordination of such a rotation. This distributed approach also places the burden of system maintenance on the participant, since NIST can do little to compensate for system errors.

The distributed model also has implications with respect to assessor fatigue and learning effects. The ability for participants to deploy arbitrarily-complex QA interfaces suggests that three-minute interactions may not be sufficient. However, since the NIST assessor interacts with *all* systems, any increase in interaction duration per system will accelerate assessor fatigue. For an evaluation with twenty runs (anticipating growth in participation), 3 minute interactions translate into 60 minutes per topic, which can reasonably be done within one sitting. Increasing the length, say, to 5 minutes translates into 100 minutes total per topic, which will tax the concentration of an assessor. Anything longer must necessarily span multiple sessions, thus introducing even more uncontrollable factors. In addition to fatigue, we must also account for learning effects. In the distributed model, as well as the previous model, interactions are not independent—what an assessor learns while interacting with one system likely impacts how he interacts with subsequent ones. Longer interaction times will only make these issues more acute. While a proper rotation will distribute this learning bias across sites, it will not eliminate such effects.

After discussions at the TREC 2006 workshop, a survey of potential participants indicated support for this distributed approach to interactive QA evaluation. Thus, the ciQA task in TREC 2007 will adopt a completely unrestricted Web-based approach to interactive QA. There are obviously many details and issues that will need to be discussed and resolved, which will provide track participants with an opportunity to chart the future of interactive QA.

## 7 Acknowledgements

We thank Hoa Dang and Ellen Voorhees for helpful discussions and their hard work in making TREC possible. We'd also like to thank participants at the TREC 2006 workshop for their valuable comments. The second author would like to thank Esther and Kiri for their kind support.

## References

J. Allan. 2005. HARD track overview in TREC 2005: High accuracy retrieval from documents. In *TREC 2005*.

H. Dang, J. Lin, and D. Kelly. 2006. Overview of the TREC 2006 question answering track. In *TREC 2006*.

S. Dumais and N. Belkin. 2005. The TREC interactive tracks: Putting the user into search. In Voorhees and Harman, editors, *TREC: Experiment and Evaluation in Information Retrieval*. MIT Press, Cambridge, MA.

W. Hersh and P. Over. 2001. Interactivity at the Text Retrieval Conference (TREC). *IP&M*, 37(3):365–367.

W. Hersh, A. Cohen, J. Yang, R. Bhupatiraju, P. Roberts, and M. Hearst. 2005. TREC 2005 Genomics Track overview. In *TREC 2005*.

J. Lin and D. Demner-Fushman. 2006. Will pyramids built of nuggets topple over? In *HLT/NAACL 2006*.

A. Spink. 1997. Study of interactive feedback during mediated information retrieval. *JASIS*, 48(5):382–394.

E. Voorhees and H. Dang. 2005. Overview of the TREC 2005 question answering track. In *TREC 2005*.

E. Voorhees. 2003. Overview of the TREC 2003 question answering track. In *TREC 2003*.