TREC 2007 CiQA Track at RMIT and CSIRO

Mingfang Wu *Andrew Turpin* *Ross Wilkinson*
Falk Scholer *Yohannes Tsegay* CSIRO ICT Centre
School of Computer Science and IT Canberra, Australia
RMIT University, Australia

1. Overview

As part of our participation in the 2007 CiQA track, the RMIT and CSIRO team investigated the following three research questions:

1. What contextual words are helpful in improving answer quality?
2. Given two answer lists of different quality, which list would a user prefer?
3. Would a user's preference choice be correlated with her own relevance judgement of an individual list?

To explore these questions, we submitted:

- Four system runs with various query formulation strategies;
- Two interactive runs, with one interface for the preference choice, and the other one for the relevance judgement of each answer sentence from an answer list.

2. Experiments

2.1. Two initial runs

We used the Indri index and search tools from the Lemur toolkit for all our system runs. When the collection was indexed, words were stemmed using the Krovetz stemmer, and words from the stoplist were removed. We chose a language model with Jelinek-Mercer smoothing ($\sigma = 0.5$) to weight and rank documents (Zhai and Lafferty, 2001).

To get an answer list for a question, we extracted a query from the topic field, retrieved the set of top 20 documents, then parsed these documents into sentences. Sentences were ranked according to each sentence's score, calculated as a combination of the longest span of matched query words, the number of matched query words, and the number of matched distinct query words.

The only difference between all our system runs lies in the way in which the query was constructed. In our baseline system run, *rmitrun1*, we used those words within the brackets embedded in a question template as a query. In our second system run, *rmitrun2*, we added some additional words into the query; namely, those words form the narrative field of a question topic, except the introduction part (such as "The analyst would like to know of"). These words give elaborated information about what is counted as an answer. For the purpose of exploring the second research question, we would like to get an answer list from *rmitrun2* that is sufficiently different from the one from *rmitrun1*, but not too dramatically dissimilar. We therefore experimented with giving different weights to the two sets of words from the title and narrative fields. Our submitted run used equal weight for the words from two fields.

2.2. Two interactive runs

To explore the second and third research questions, we set up two interfaces: one for preference choice between two lists, and the other for relevance judgement of an individual list. In the interface for the preference choice (submission run-id: *rmitrun3)*, for each question, we took two answer lists each from *rmitrun1* and *rmitrun2* and then showed the two lists side by side, as in Figure 1. The two answer lists from *rmitrun1* and *rmitrun2* were randomly assigned to the left or the right panel in each question, but overall, the two systems have equal chance to be on either side for 30 questions. Users at NIST were

required to browse the two answer lists, then select their preferred answer list by clicking the button located on top of each list, and finally fill in a questionnaire (as shown in Figure 2). Assessors were given five minutes for each question. In our systems, assessors/users were reminded to move on to the questionnaire by the end of four minutes. Because of this time limit, only the top ten answer sentences were displayed for each question.

This comparison of two systems through preference choice was first introduced and tested by Thomas and Hawking (Thomas and Hawking, 2006). By using a similar interface with two panels, they evaluated two alternate search systems. Using both supplied queries and their own real queries, the participants found no discernable left-right bias, and subjects were able to reliably distinguish between high- and low-quality result sets. Therefore, the use of such a comparison study can avoid many of the costs and biases of familiar evaluation methods. We adopted this interface to test if this finding still holds true for the complex question answering task, and importantly we used the questionnaire instrument to gather information about why they chose one list over another.

Our third research question is: would this preference choice be correlated with a user's own relevance judgement of an individual list? In our system *rmitrun4*, the same lists that were used in *rmitrun3* were displayed in a single panel (Figure 3), and assessors were asked to make a relevance judgement for each answer sentence. We also set up a time reminder, as in *rmitrun3*; assessors were then required to fill in the questionnaire (Figure 4) to give their overall assessment of a list. Ideally, for each question, both lists from these two systems should be judged, so we can compare the preference choice with the relevance judgement of individual lists on a question by question basis. However, in our second interface, a question could only appear once. Therefore, we had to make a compromise by displaying answer lists from each system for only half of the questions. When questions were assigned to systems, the question types were considered, so questions of the same type (such as "financial relationship") would have a roughly equal chance from both systems. In this way, we could only do the comparison on a system by system basis.

2.3. Two final runs

We submitted two runs for the final submission. The run *rmitrun5* is a relevance feedback run based on the relevance judgments that were collected from the interface *rmitrun4*. In this run, a query includes the words inside brackets from question template (as in the run *rmitrun1*), as well as words from sentences (up to five) that were judged as "definitely an answer" by users of *rmitrun4*. If there isn't any sentence judged as "definitely an answer" for a question, sentences (up to five) that were judged "not sure, need to read original documents" would be chosen (question 72 and 77 from *rmitrun2* and question 66 and 73 from *rmitrun1* fall in this situation). If all sentences were judged as "definitely not an answer" (the question 68), then the top five ranked sentences would be taken.

In the run *rmitrun6*, we used the following steps to get an initial 20 candidate documents:

1. Take words from within each bracket, treat them as a phrase, and use Boolean "and" to connect each phrase. For example, for the question 56:
   "What evidence is there for transport of [illegal immigrants] from [Croatia] to [the European Union]?"
   The query is: "illegal immigrants" and "Croatia" and "European Union"
2. Relax the above queries by removing Boolean "and";
3. Relax the above queries by removing quotation marks.

A search will be stopped at a stage when a list of twenty candidate documents have been retrieved at that point, otherwise the top ranked documents form the next step search would be used to make up the list. As a result, there were 13/8/9 of questions got their lists of 20 documents at steps 1/2/3 respectively.

3. Results

3.1. System runs

Overall, the Pyramid F scores for four system runs are very close to each other: there is no statistically significant difference between any runs. In fact, the baseline run *rmitrun1* is slightly better than the other three runs. The three alternative querying strategies improved less topics (10/10/7 for rmitrun2/5/6) but worsen more topics (16/19/9 for rmitrun2/5/6). The run *rmitrun2* was not expected to be better than *rmitrun1*, the weights of different query components were chosen to separate the corresponding lists of a question from two runs.

Question	Rmitrun1	Rmitrun2	Rmitrun5	Rmitrun6
56	0.378	0.379	0.236	0.332
57	0.516	0.510	0.134	0.505
58	0.568	0.487	0.564	0.568
59	0.674	0.633	0.680	0.635
60	0.316	0.447	0.560	0.492
61	0.317	0.057	0.291	0.309
62	0.061	0.230	0.174	0.061
63	0.206	0.369	0.200	0.174
64	0.272	0.106	0.400	0.270
65	0.206	0.184	0.000	0.098
66	0.083	0.446	0.000	0.000
67	0.528	0.539	0.528	0.378
68	0.000	0.210	0.000	0.000
69	0.402	0.272	0.299	0.244
70	0.508	0.468	0.560	0.564
71	0.564	0.566	0.454	0.498
72	0.581	0.482	0.602	0.307
73	0.217	0.229	0.291	0.182
74	0.393	0.393	0.274	0.451
75	0.391	0.369	0.262	0.458
76	0.150	0.205	0.068	0.106
77	0.466	0.527	0.468	0.404
78	0.638	0.628	0.631	0.504
79	0.275	0.407	0.388	0.463
80	0.383	0.369	0.422	0.450
81	0.222	0.162	0.140	0.109
82	0.245	0.112	0.255	0.197
83	0.298	0.238	0.399	0.324
84	0.474	0.42	0.474	0.419
85	0.625	0.374	0.543	0.477
Average	0.365	0.361	0.343	0.333

Table 1. Pyramid F scores for our system runs

3.2. Interactive Runs

3.2.1. Preference choice

There are 30 questions in total. No preference was made for questions 70 and 78. Among the remaining 28 questions, 16 of them were chosen with a convincing reason that one list was better than another - *rmitrun1* and *rmitrun2* each got 11 votes and 5 votes respectively. For the remaining 12 questions, assessors just made a random (but not side-biased) choice and commented later that in fact they thought there was not any difference between the two lists.

To explore if an assessor indeed choose a list that is of high quality, we compared an assessor's preference choice with the official nugget pyramid evaluation that was aggregated from nine assessors. Considering that the assessors/users of our interactive interfaces saw only the top ten answer sentences for each question, we adopt nugget precision to evaluate the quality of shown lists. As in Dang, Lin and Kelly

(2006), we approximate this nugget precision by a length allowance based on the number of both vital and okay nuggets, that is:

length: # of non-whitespace characters in the entire answer string
okay: # of okay nuggets returned in a response
vital: # of vital nuggets returned in a response
allowance (a) = 100 * (okay + vital)

$$\text{Precision} = \begin{cases} 1 & \text{if } length < a \\ 1 - \dfrac{length - a}{length} & \text{otherwise} \end{cases}$$

Thus, this measure would tell us how much useful information is contained in a string of certain length. However, similar to the precision in a document search task, this measure doesn't give information on whether the useful information is at the top or bottom of a list. We then calculated average nugget precision as in a document search task, but used the above precision formula to approximate the mean of the precision after each relevant answer sentence (a sentence that has either okay or vital nuggets). The correlations between nugget precision and average nugget precision are 0.57 for rmitrun1 and 0.83 for rmitrun2.

Figure 5 shows the relationship between the average nugget precision and the preference choices of the assessors. An assessor's choice is marked after the question number. The choice labels A, B, E and X represent rmitrun1, rmitrun2, no difference, and no record of choices, respectively. The tick and cross symbols indicate if an assessor agreed or disagreed with the quality evaluation of the lists.

It can be seen that there is a big difference between a user's preference and the average nugget precision: the ratio for disagreement to agreement is 20:8. (A similar ratio exists between the preference and the nugget precision.) The average nugget difference between some paired lists may be too small to be recognised by the users, even through there still exists the disagreement for 10 and 6 questions, with each number corresponding to the average nugget difference at the 0.1 and 0.2 level. This result is quite different from Thomas and Hawking's work where users were asked to compare two lists for the same query but for a document relevance judgement task.

3.2.2. Answer identification

In our second interactive interface (*rmitrun4*), users were asked to identify whether a presented answer sentence indeed contains an answer on a three-level semantic scale, namely: "definitely an answer", "not sure, need to read original document", and "definitely not an answer". We compared this judgement with the final judgement made by nine NIST assessors (it is believed this includes the users who interacted with our interfaces). We found that out of 300 answer sentences (30 questions x 10 sentences each), the assessors of our interfaces disagreed with the group judgement for 111 sentences. This difference may be one of the reasons that caused the big difference between the preference choice and the (average) nugget precision reported above.

4. Discussion

We tested various querying strategies for the system runs. Overall, we didn't see significant improvement by using any of these tested query formulation strategies. A more thorough analysis of questions and the performance of these runs on a question by question basis is required.

For the interactive runs, we observed a big difference between the preference choice and the comparison of two lists (using the average nugget precision measure), as well as a big difference in answer nugget identification between individual assessors of our interactive interfaces and a group of assessors as a whole. This again raises the classic questions about relevance: who make the relevance judgement, in which circumstances are the relevance judgements made, and how should relevance be measured? These issues

are very important for the interactive evaluation of information retrieval systems, as we often don't see a similarity between system performance and user performance.

References

Lemur project: http://www.lemurproject.org

H. T. Dang and J. Lin and D. Kelly. Overview of the TREC 2006 Question Answer Track. *Proceedings of the Fifteenth Text REtrieval Conference (TREC 2006)*, November 2006, Gaithersburg, Maryland.

P. Thomas and D. Hawking, Evaluation by Comparing Result Sets in Context, *Proceedings of CIKM'06*. Virginia USA, pp.94-101, 2006

C. Zhai and J. Lafferty. A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In *Proceedings of the 24st International ACM-SIGIR Conference on Research and Development in Information Retrieval*. Louisianna USA, pp.334-342, 2001

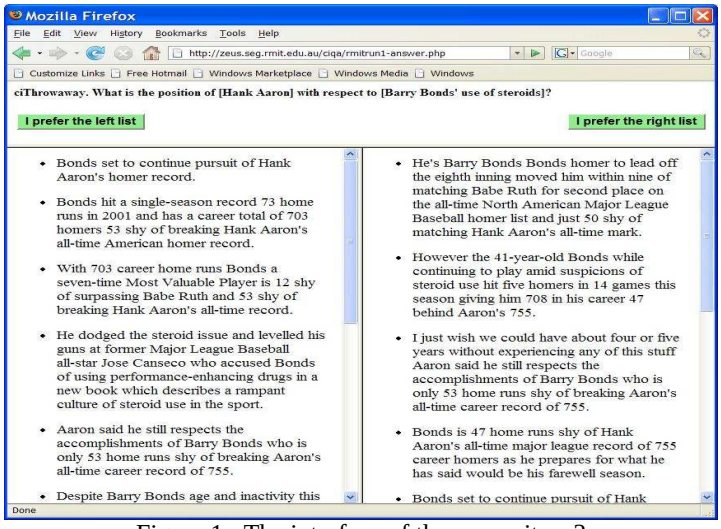

Figure 1: The interface of the run rmitrun3

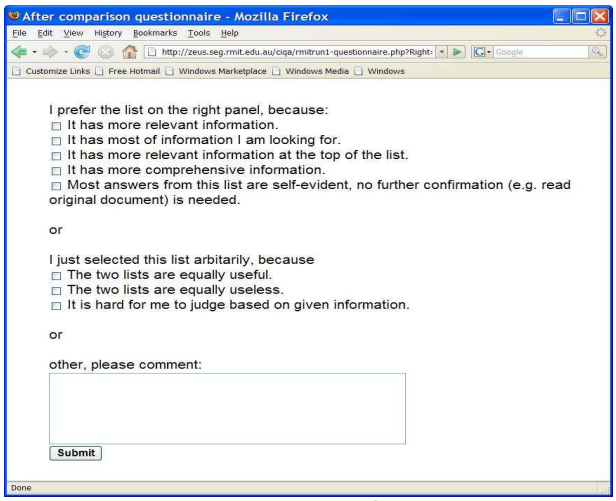

Figure 2. The questionnaire of the run rmitrun3

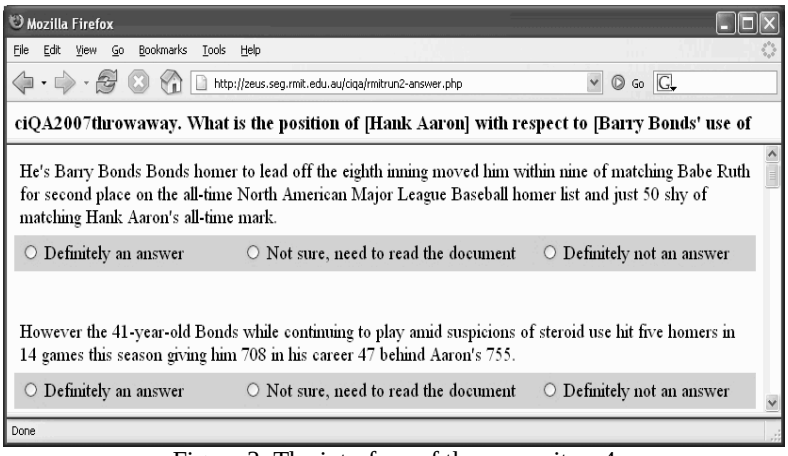

Figure 3. The interface of the run rmitrun4

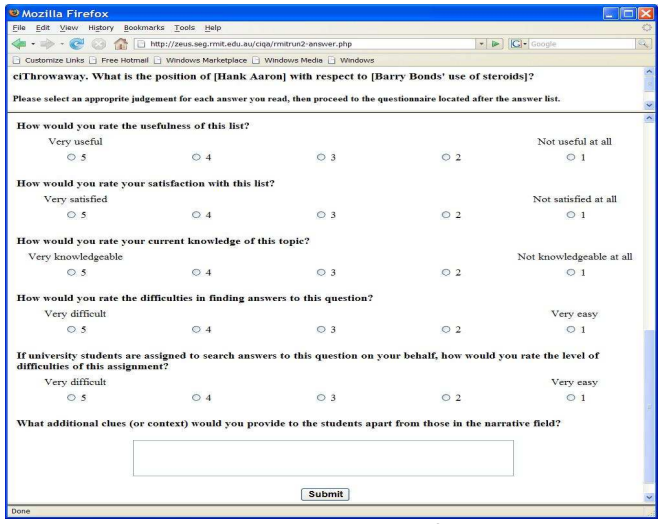

Figure 4. The questionnaire of rmitrun4

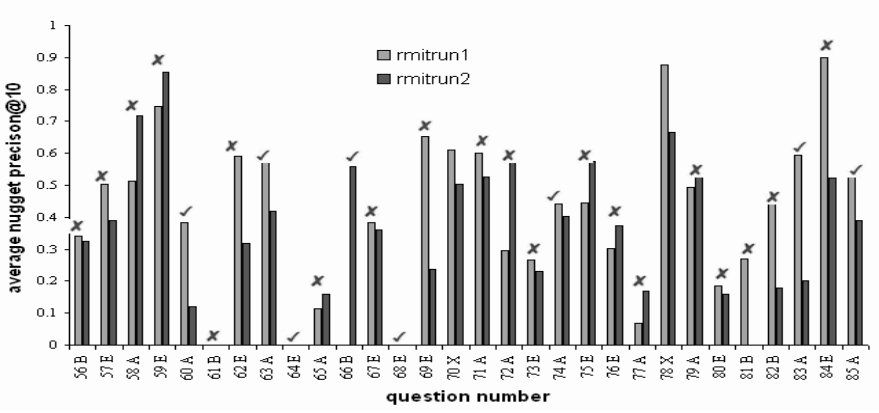

Figure 5. The correlation between preference choice and average precision
(Choice A: rmitrun1, Choice B: rmitrun2, Choice E: no difference, X: no record of choice)