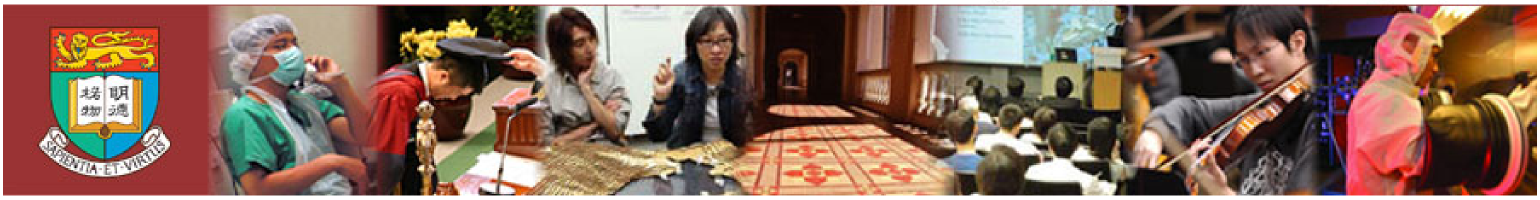


Title	Building on Redundancy: Factoid Question Answering, Robust Retrieval and the "Other"
Author(s)	Roussinov, D; Filatova, E; Chau, MCL; Robles-Flores, JA
Citation	The 14th Text REtrieval Conference (TREC 2005), Gaithersburg, MD, 15-18 November 2005
Issued Date	2005
URL	http://hdl.handle.net/10722/112219
Rights	This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.

Building on Redundancy: Factoid Question Answering, Robust Retrieval and the “Other”

Dmitri Roussinov
Arizona State University
Department of Information Systems
Dmitri.Roussinov@asu.edu

Michael Chau
School of Business
The University of Hong Kong
mchau@business.hku.hk

Elena Filatova
Columbia University
Department of Computer Science
filatova@cs.columbia.edu

José Antonio Robles-Flores
Arizona State University/ESAN
Department of Information Systems
Jose.Robles@asu.edu

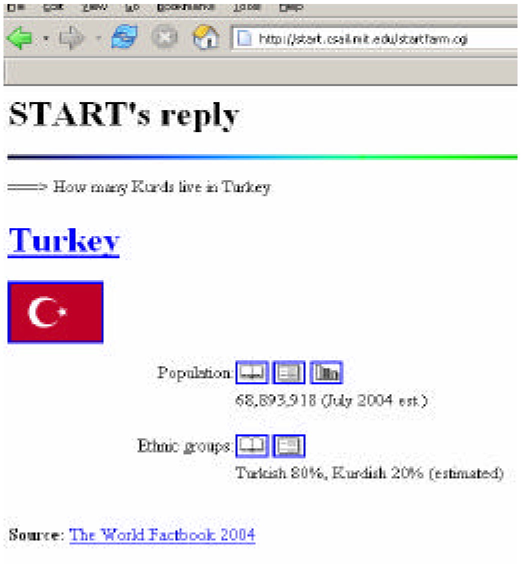
ABSTRACT

We have explored how redundancy based techniques can be used in improving factoid question answering, definitional questions (“other”), and robust retrieval. For the factoids, we explored the meta approach: we submit the questions to the several open domain question answering systems available on the Web and applied our redundancy-based triangulation algorithm to analyze their outputs in order to identify the most promising answers. Our results support the added value of the meta approach: the performance of the combined system surpassed the underlying performances of its components. To answer definitional (“other”) questions, we were looking for the sentences containing re-occurring pairs of noun entities containing the elements of the target. For robust retrieval, we applied our redundancy based Internet mining technique to identify the concepts (single word terms or phrases) that were highly related to the topic (query) and expanded the queries with them. All our results are above the mean performance in the categories in which we have participated, with one of our robust runs being the best in its category among all 24 participants. Overall, our findings support the hypothesis that using as much as possible textual data, specifically such as mined from the World Wide Web, is extremely promising.

FACTOID QUESTION ANSWERING

The Natural Language Processing (NLP) task, which is behind Question Answering (QA) technology, is known to be Artificial Intelligence (AI) complete: it requires the computers to be as intelligent as people, to understand the deep semantics of human communication, and to be capable of common sense reasoning. As a result, different systems have different capabilities. They vary in the range of tasks that they support, the types of questions they can handle, and the ways in which they present the answers.
By following the example of meta search engines on the Web (Selberg & Etzioni, 1995), *we advocate combining several fact seeking engines into a single “Meta” approach*. Meta search engines (sometimes called metacrawlers) can take a query consisting of keywords (e.g. “*Rotary engines*”), send them to several portals (e.g. Google, MSN, etc.), and then combine the results. This allows them to provide better coverage and specialization. The examples are MetaCrawler (Selberg & Etzioni, 1995), 37.com (www.37.com), and Dogpile (www.dogpile.com). Although, the keyword based meta search engines have been suggested and explored in the past, we are not aware of the similar approach tried for the task of open domain/corpus question answering (fact seeking).

The practical benefits of the meta approach are justified by general consideration: eliminating “weakest link” dependency. *It does not rely on a single system which may fail or may simply not be designed for a specific type of tasks (questions).* The meta approach promises *higher coverage and recall of the correct answers* since different QA engines may cover different databases or different parts of the Web. In addition, the meta approach *can reduce subjectivity* by querying several engines; like in the real-world, one can gather the views from several people in order to make the answers more accurate and objective. The speed provided by several systems queried in parallel can also significantly exceed those obtained by working with only one system, since their responsiveness may vary with the task and network traffic conditions. In addition, the meta approach fits nicely into a becoming-popular Web services model, where each service (QA engine) is independently developed and maintained and the meta engine integrates them together, while still being organizationally independent from them. Since each engine may be provided by a commercial company interested in increasing their advertising revenue or a research group showcasing their cutting edge technology, the *competition mechanism will also ensure quality and diversity* among the services. Finally, a meta engine can be *customized* for a particular portal such as those supporting business intelligence, education, serving visually impaired or mobile phone users.


File Edit View Go Bookmarks Tools Help
http://start.csail.mit.edu/startfarm.cgi

START's reply

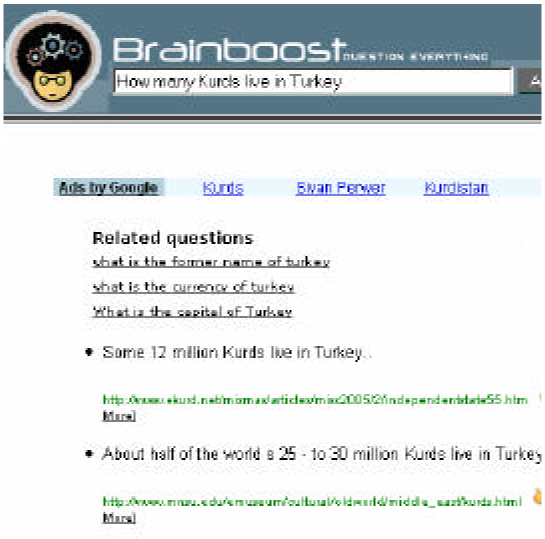
==> How many Kurds live in Turkey

Turkey

Population:
68,893,918 (July 2004 est.)

Ethnic groups:
Turkish 80%, Kurdish 20% (estimated)

Source: The World Factbook 2004



Related questions
what is the former name of turkey
what is the currency of turkey
What is the capital of Turkey

- Some 12 million Kurds live in Turkey..

  http://www.ekurd.net/mismas/articles/misc2005/2/independentstate55.htm
  More!

- About half of the world s 25 - to 30 million Kurds live in Turkey

  http://www.mnsu.edu/emuseum/cultural/oldworld/middle_east/kurds.html
  More!

Figure 1. Example of START output. **Figure 2. Example of Btainboost output.**

Meta Approach Defined

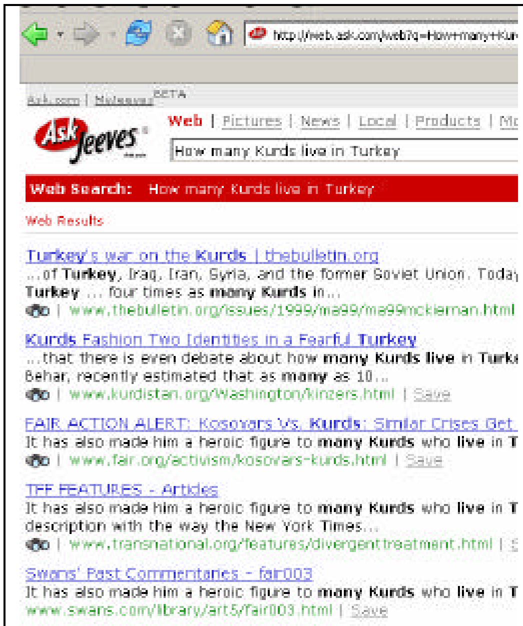
We define *a fact seeking meta engine* as the system that can combine, analyze, and represent the answers that are obtained from several underlying systems (called *answer services* throughout our paper). At least some of these underlying services (systems) have to be capable of providing *candidate answers* to some types of questions asked in a natural language form, otherwise the overall architecture would not be any different from a single fact seeking engine which are typically based on a commercial keyword search engines, e.g. Google. The technology behind each of the answer services can be as complex as deep semantic NLP or as simple as shallow pattern matching.

Fact Seeking Service	**Web address**	**Output Format**	**Organization/System**	**Performance in our evaluation (MRR)**
START	start.csail.mit.edu	Single answer sentence	Research Prototype	0.049**
AskJeeves	www.ask.com	Up to 200 ordered snippets	Commercial	0.397**
BrainBoost	www.brainboost.com	Up to 4 snippets	Commercial	0.409*
ASU QA on the Web	qa.wpcarey.asu.edu	Up to 20 ordered sentences	Research Prototype	0.337**
Wikipedia	en.wikipedia.org	Narrative	Non profit	0.194**
ASU Meta QA	**http://qa.wpcarey.asu.edu/**	**Precise answer**	**Research Prototype**	**0.435**

Table 1. The fact seeking services involved, their characteristics and performances in the evaluation on the 2004 questions. * and ** indicate 0.1 and .05 levels of statistical significance of the difference from the best accordingly.

Challenges Faced and Addressed

Combing multiple fact seeking engines also faces several challenges. First, *the output formats of them may differ*: some engines produce exact answer (e.g. START), some other present one sentence or an entire snippet (several sentences) similar to web search engines, as shown in Figures 1-4. Table 1 summarizes those differences and other capabilities for the popular fact seeking engines. Second, *the accuracy of responses may differ* overall and have even higher variability depending on a specific type of a question. And finally, we have to *deal with multiple answers*, thus removing duplicates, and resolving answer variations is necessary. The issues with merging search results from multiple engines have been already explored by MetaCrawler (Selberg & Etzioni, 1995) and fusion studies in information retrieval (e.g. Vogt & Cottrell, 1999) but only in the context or merging lists of retrieved text documents. We argue that *the task of fusing multiple short answers, which may potentially conflict or confirm each other, is fundamentally different and poses a new challenge for the researchers*. For example, some answer services (components) may be very precise (e.g. START), but cover only a small proportion of questions. They need to be backed up by less precise services that have higher coverage (e.g. AskJeeves). However, backing up may easily result in diluting the answer set by spurious (wrong) answers. Thus, *there is a need for some kind of triangulation of the candidate answers provided by the different services or multiple candidate answers provided by the same service.*


http://web.ask.com/web?q=How+many+Ku
Ask.com | MyJeeves BETA
Ask Jeeves
Web | Pictures | News | Local | Products | Mo

How many Kurds live in Turkey

Web Search: How many Kurds live in Turkey

Web Results

Turkey's war on the Kurds | thebulletin.org
...of Turkey, Iraq, Iran, Syria, and the former Soviet Union. Today
Turkey ... four times as many Kurds in...
| www.thebulletin.org/issues/1999/ma99/ma99mckiernan.html

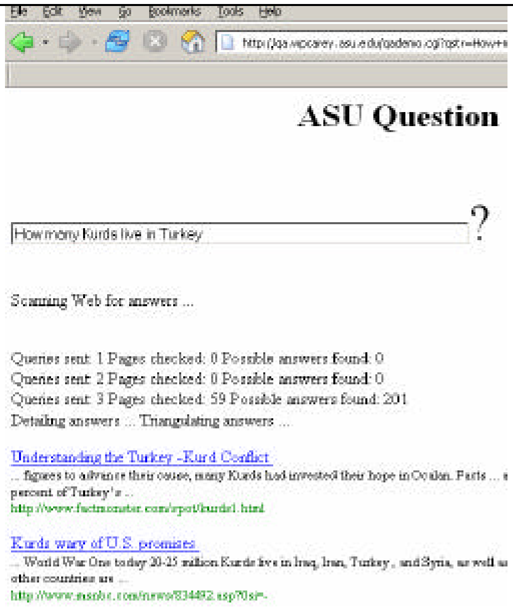
Kurds Fashion Two Identities in a Fearful Turkey
...that there is even debate about how many Kurds live in Turk
Behar, recently estimated that as many as 10...
| www.kurdistan.org/Washington/kinzers.html | Save

FAIR ACTION ALERT: Kosovars Vs. Kurds: Similar Crises Get
It has also made him a heroic figure to many Kurds who live in T
| www.fair.org/activism/kosovars-kurds.html | Save

TFF FEATURES - Articles
It has also made him a heroic figure to many Kurds who live in T
description with the way the New York Times...
| www.transnational.org/features/divergenttreatment.html | S

Swans' Past Commentaries - fair003
It has also made him a heroic figure to many Kurds who live in T
www.swans.com/library/art5/fair003.html | Save

Figure 3. Example of Ask Jeeves output.


File Edit View Go Bookmarks Tools Help
http://qa.wpcarey.asu.edu/qademo.cgi?qstr=How+

ASU Question

How many Kurds live in Turkey ?

Scanning Web for answers ...

Queries sent: 1 Pages checked: 0 Possible answers found: 0
Queries sent: 2 Pages checked: 0 Possible answers found: 0
Queries sent: 3 Pages checked: 59 Possible answers found: 201
Detailing answers ... Triangulating answers ...

Understanding the Turkey -Kurd Conflict
... figures to advance their cause, many Kurds had invested their hope in Ocalan. Paris ... percent of Turkey's ...
http://www.factmonster.com/spot/kurds1.html

Kurds wary of U.S. promises
... World War One today 20-25 million Kurds live in Iraq, Iran, Turkey, and Syria, as well as other countries are ...
http://www.msnbc.com/news/834492.asp?0si=-

Figure 4. Example of ASU QA output.

Triangulation, a term which is widely used in intelligence and journalism, stands for confirming or disconfirming facts, by using multiple sources. Roussinov et al. (2004) went one step further than using the frequency counts explored earlier by Dumais et al. (2002) and groups involved in TREC competitions. They explored a more fine-grained triangulation process which we also used in our prototype. Their algorithm can be demonstrated by the following intuitive example. Imagine that we have two candidate answers for the question *"What was the purpose of the Manhattan Project?":* 1) *"To develop a nuclear bomb"* 2) *"To create an atomic weapon"*. These two answers support (triangulate) each other since they are semantically similar. However, a straightforward frequency count approach would not pick this similarity. The advantage of triangulation over simple frequency counting is that it is more powerful for less "factual" questions, such as those that may allow variations in the correct answers.

In order to enjoy the full power of triangulation with factoid questions (e.g. *Who is the CEO of IBM*?), the candidate answers have to be extracted from their sentences (e.g. *Samuel Palmisano*), so they can be more accurately compared with the other candidate answers (e.g. *Sam Palmisano*). That is why *the meta engine needs to possess answer understanding capabilities as well, including such crucial capability as question interpretation and semantic verification of the candidate answers* to check that they belong to a desired category (*person* in the example above).

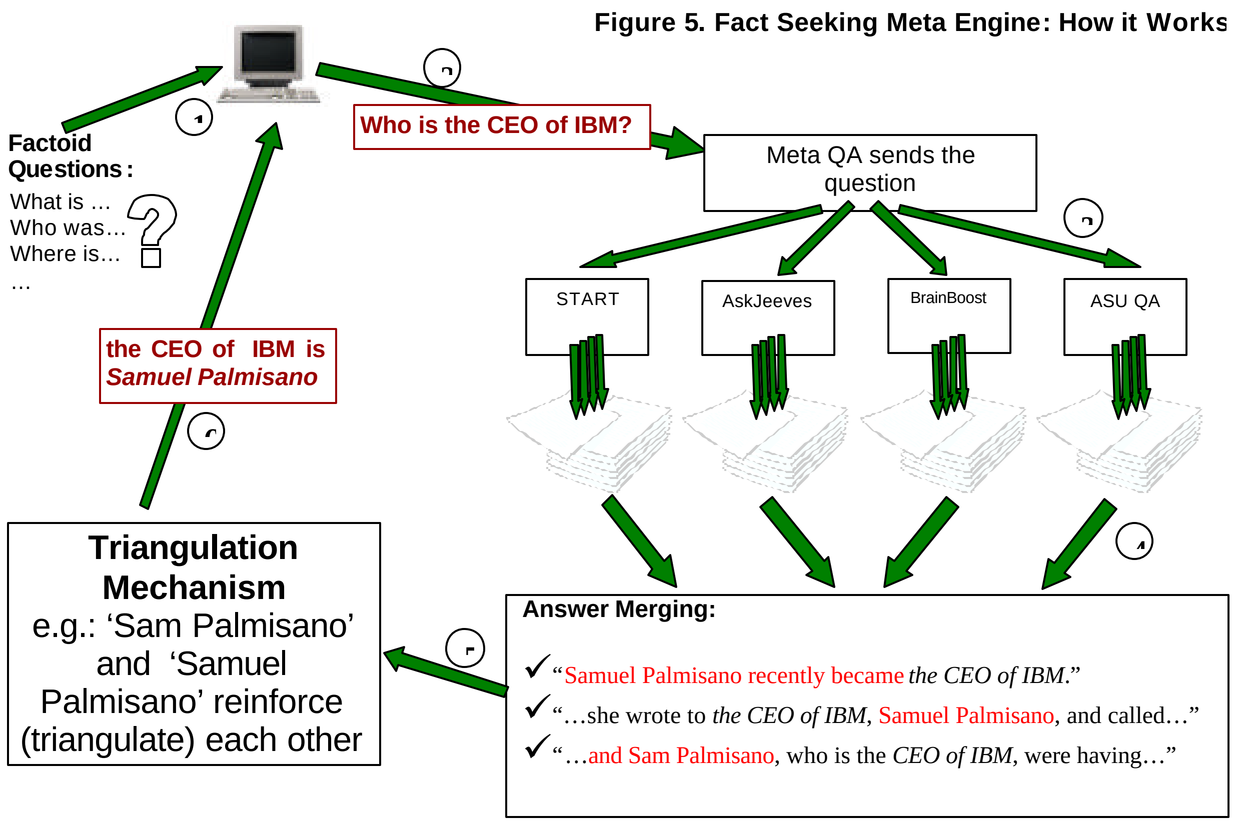


Figure 5. Fact Seeking Meta Engine: How it Works
Factoid Questions:
What is …
Who was…
Where is…
…
Who is the CEO of IBM?
Meta QA sends the question
START
AskJeeves
BrainBoost
ASU QA
the CEO of IBM is *Samuel Palmisano*
Triangulation Mechanism
e.g.: 'Sam Palmisano' and 'Samuel Palmisano' reinforce (triangulate) each other
Answer Merging:
✓ "Samuel Palmisano recently became *the CEO of IBM*."
✓ "…she wrote to *the CEO of IBM*, Samuel Palmisano, and called…"
✓ "…and Sam Palmisano, who is the *CEO of IBM*, were having…"

Figure 5. The Meta approach to fact seeking.

Fact Seeking Engine Meta Prototype: Underlying Technologies and Architecture

In the first version of our prototype, we included several freely available demonstrational prototypes and popular commercial engines on the Web that have some QA (fact seeking) capabilities, specifically START, AskJeeves, BrainBoost and ASU QA (Table 1, Figures 1-4). We also added Wikipedia to the list. Although it does not have QA capabilities, it provides good quality factual information on a variety of topics, which adds power to our triangulation mechanism. Google was not used directly as a service but BrainBoost and ASU QA are already using it among the other major keyword search engines. The meta-search part of our system was based on the MetaSpider architecture (Chau et al., 2001; Chen et al., 2001). Multi-threads are launched to submit the query to fetch the candidate answers from each service. After these results are obtained, the system performs answer extraction, triangulation and semantic verification of the results, based on the algorithms from Roussinov et al. (2004). Figure 5 summarizes the overall process. For the TREC competition, we applied the answer projection algorithm, same as last year, that tried to find the best supporting document within the TREC collection (Aquaint) by matching the words from the question and the target.

We have been maintaining a working prototype on the web (http://qa.wpcarey.asu.edu/) since August 2004 and have already accumulated 1000+ questions that we can use to test our future research hypothesis and fine-tune our algorithms. The prototype has been featured in Information Week (Claburn, 2005) as one of the promising directions in the "Web Search of Tomorrow."

Testing on 2004 questions

Before the answer submission deadline this year, we fine-tuned the weights given to the underlying answer services and evaluated our meta approach. We used the set of 200 test questions and regular expression answer keys from the Question-Answering Track of the TREC 2004 conference (Voorhees and Buckland, 2004). Although various metrics have been explored in the past, we used mean reciprocal rank (MRR) of the first correct answer as in the TREC-s 2001, 2002 and in Dumais et al. (2002). This metric assigns a score of 1 to the question if the first answer is correct. If only the second answer is correct, the score is ½, the third correct results in 1/3, etc. The drawback of this metric is that it is not the most sensitive since it only considers the first correct answer, ignoring what follows. However, it is still more sensitive than the TREC 2004 and 2005 official metrics that only look at the first answer. We did not use the “degree of support” of the answer within the document as part of the metric due to its known difficulty (Lin, 2005), and thus only checked if the answer was correct, which is sometimes called “lenient” evaluation, to which the concerns of Lin et al. do not apply.

Co-reference resolution

We used the same heuristic rules as last year (Roussinov et al., 2004) to resolve the necessary co-references, similarly to how it was done by most participants. Specifically, we replaced the pronouns with the targets. (E.g. converting “How fast can it fly?” into “How fast can F16 fly?”). The target was also appended at the end of the question if it was not already present in the question. The approach worked correctly in 75% of the cases. Most errors were caused by events (new this year). However, our approach happened to be resilient to those types of question interpretation errors since it was based on the redundancy rather than on the accurate interpretation. Consider the following example: Target = “Miss Universe 2000 crowned”, Question = “What country did the winner represent?” The correct interpretation would be rather difficult at the current state of the art of AI since it would require knowing that “Miss Universe” event is supposed to have a winner, finding the winner, and substituting the winner name to the original question. However, for our redundancy based approach it was not essential. The question that we sent to the answering services was “What country did the winner represent Miss Universe 2000 crowned?” Although the question sound awkward, the underlying services still returned plenty of snippets related to “Miss Universe 2000.” Since the beginning of the question matched the pattern “What \T did \Q \V” (where T = “country”, \Q = “the winner” and \V = “represent”) our semantic verification mechanism was deliberately looking for mentioning of countries, but not the other types such as people names, dates, etc. This often resulted in correct answers.

Official Runs Submitted

We submitted two official runs: 1) **ASUQA01** that used only Google as underlying source of answers and was not essentially different from the system used last year, and 2) **ASUQA02** that used our meta engine, which was essentially equivalent to using our last year answer extraction and triangulation engine on the answers obtained from the underlying answer services listed in Table 1. As we expected, *ASUQA02 was significantly better than ASUQA01in the measured accuracy : 0.180 vs. 0.149 . It was also within 20% results across all participants (0.152)*.

Our finding corroborates the findings in the more general domain of web searching, in which meta-approach results in better coverage than each individual search engine. In the case of using QA technology, which is known to be very knowledge intensive and expensive to create, it is extremely challenging and also important for the meta engine to be at least as good as the best underlying component, otherwise the correct answer can be missed and diluted by erroneous ones. As our empirical finding illustrates, the triangulation algorithms that we have employed has successfully overcome that challenge. Since our triangulation was capitalizing on the inherent redundancy on the web or among the answers, this result also testifies to the power of redundancy based techniques.

Both NLP-based and the approaches that require elaborate manually created patterns have a strong advantage: they can be applied to smaller collections (e.g. corporate repositories) and provide good performance. However, because expensive knowledge engineering is required to build such systems and possibly the entailing intellectual property issues, none of the known top performing systems has been made publicly open to the other researches for follow up investigations. As result, it is still unknown what approaches exactly work in different conditions, for example how well they would extend outside of TREC domain. On the other side, the algorithms behind the systems that do not require extensive knowledge engineering, but still demonstrate reasonable performance, have been available open to the public, e.g. (Dumais et al., 2002; Roussinov & Robles, 2004). We believe that from the research perspective, those transparent “knowledge-light” systems and approaches are no less interesting that the top commercial systems since they allow replication and independent testing by the other researchers.

DOCUMENT RANKING

The task of “document ranking,” defined for some factoid questions, was to order documents in the target collection so as to maximize the likelihood of having the answer in the top returned documents. Since we did not use the target collection to obtain the answer, but rather obtained the answer through triangulation of web answers, we only used document ranking to perform the answer projection at the end. For this, we used bm25 ranking function from Lemur 3.1[1]. The query consisted of the question, the target and the answer merged together. We submitted the obtained order as our official document ranking, same with both our runs. Our result (mean R-precision of 0.3032) was significantly above the mean across all participants

[1] http://www.lemurproject.org/

(0.1666) and ranked #6. We believe this was because we “looked ahead:” retrieved documents when already having a certain answer in mind rather than retrieving documents based only on the target and the question. This result also indicates the promise of the redundancy based approach to rank documents in order to respond to the factoid information request (question) when precise answer is not really required, which simulates many practical situations, e.g. searching the web.

ANSWERING “OTHER” QUESTIONS

“Other” Questions in TREC 2005

For the first time questions of the “other” type were studied within TREC 2004 QA evaluation. “Other” questions substituted definition questions studied within TREC 2003 (Voorhees, 2003) and “asked for additional information about the target that was not covered by previous [factoid and list questions] in the series” (Voorhees, 2004). “Other” question was asked about every target. The text snippets submitted as potential answers to the “other” question were checked whether they contained “vital” or ”okay” nuggets about the target. The vital nuggets “must appear in a definition for that definition to be good.” (Voorhees, 2003). “Non-vital nuggets act as don’t care conditions in that the assessor believes the information in the nugget to be interesting enough that returning the information is acceptable in, but not necessary for, a good response” (Voorhees, 2004). The presence in the answer of the ‘okay’ nuggets did not increase the recall and did not decrease the precision.

Our approach was almost identical to the one described in (Filatova and Hatzivassiloglou, 2003) and relied on redundancy as well. It capitalized on the fact that multiple co-occurrences of the target with the named entities within one sentence could potentially capture the most interesting (vital or OK) properties of the target.

One of the crucial innovations of the TREC 2005 QA Track was that the targets, for which the questions were formulated, included not only things, organizations and people but also events. For example, some of the event targets used in TREC 2005 were: “Miss Universe 2000 crowned”, “France wins World Cup in soccer”, and “Crash of Egypt Air Flight 990”. Clearly, answers to the “other” question for the event targets cannot rely on the patterns created to extract definition information.

In our system we did not make a distinction between event targets and targets requiring definition-related nuggets. We applied the same technique to extract potential answers to the “other” question for all the targets. Besides, we did not use any external corpora but extracted potential answers from the list of ranked documents provided by NIST. Our procedure of answer extraction had two stages: (i) gather statistics about word triplet co-occurrences from the documents provided for each target by NIST; (ii) extract text snippets corresponding to the most frequent word triplets.

Word triplets. We analyzed only those top 50 documents which were provided by NIST for each target. Using BBN IdentiFinder (Bikel et al., 1999) we identified the named entities present in those documents. We used the named entities of the following types: DATE, LOCATION, ORGANIZATION, PERSON, and TIME. We also identified all the nouns and verbs using part-of-speech tagger from Alembic Workbench (Day et al., 1997) and incorporated the ten most frequent nouns into the list of named entities. We used HFNN tag for them in the examples listed in Table 2.. From the target related documents we extracted all the pairs of named entities that appeared within one sentence. We preserved only those pairs of named entities, where at least one of the two elements was a substring of or equal to the entire target. E.g. for “1998 Nagano Olympic Games,” it was enough to contain “Nagano Olympic Games.” For each target we obtained a list of triplets consisting of two named entities (or frequent nouns) and a verb or a noun that appeared between those two. For each triplet in the list we obtained the number of its occurrences in the top 50 documents provided by NIST for the target. Table 2 presents two triplets extracted for the target “OPEC”. Table 3 presents two triplets extracted for the target “1998 Nagano Olympic Games.”

Table 2. A sample of word triplets for target 128 "OPEC".

Count	Triplet
21	price/[HFNN] - barrel/NN - OPEC/[ORGANIZATION]
11	OPEC/[ORGANIZATION] - world/NN - oil/[HFNN]

Table 3. A sample of word triplets for target 96 "1998 Nagano Olympic Games".

Count	Triplet
7	City/[HFNN] - host/VB - Games/[HFNN]
5	Nagano/[LOCATION] - get/VB - games/[HFNN]

Sentence selection. We preserved only the triplets that occurred more than once in the top 50 documents provided by NIST for the target. For each triplet, we chose the longest sentence containing this triplet. If the sentence was longer than 250 characters then we truncated it at the nearest punctuation mark or the beginning of the sentence to the left of the first elements of the triplet; and for the nearest punctuation mark or the end of the sentence to the right of the third elements of the triplet.

2.3 Official Results and Discussion

Though all the text snippets in our output were extracted according to different triplets, some of the text snippets contained similar (repeating) information. E.g. we output "Japan's most famous film director, Akira Kurosawa, died at his home Sunday at the age of 88, Kyodo news agency reported." and "Japan's internationally renowned film director Akira Kurosawa died Sunday at age 88." However, irrespectively of how many times a nugget judged "vital" or "OK" was repeated (e.g. "Kurosawa died at age 88"), it received credit only once, which resulted in low precision because the systems were penalized for the total length of all returned snippets. On the other side, the recall of our system was high. We believe this is because the important information was indeed typically repeated in the target collection. Overall, our technique worked well: the average F-measure for our system (0.171) was above the median average (0.156). This shows the promise of a simple redundancy based approach to answering definitional ('other') TREC questions.

ROBUST RETRIEVAL

Past experience with TREC topics indicates that while the query expansion based on blind feedback (adding the terms form the top returned documents to the query) is the most effective way of improving performance, it is not effective on the worst topics due to a phenomenon commonly known as "query drift:" when the top documents are irrelevant, so are the added terms.

Since this year topics have been selected from the worst prior year topics, we were deliberately looking into different expansion strategies. Inspired by the success of our (Roussinov & Zhao, 2003) and other researchers' (Kwok et al., 2004) work on Internet mining, we developed a method called Context Specific Similarity Discovery (CSSD), the details of which can be found in (Roussinov et al., 2005). The idea behind the method is to identify the concepts (single word terms or phrase) that are highly related to the topic (query). We believe using the Internet for this purpose provides much more data for statistical analysis compared to using only the target collection itself (e.g. Aquaint) when it is implemented using blind (pseudo) relevance feedback and its variants.

In this year TREC, we submitted our query to Google and built so called *Internet language model* for it. We designed and trained a special formula for the probability of being related to the topic using logistic regression and proceeded through the following steps:

Step 1. The proper combination of the title and description was merged into a single query and sent to Google.

Step 2. The full text of the top 200 pages returned by Google was downloaded as the *mining corpus* (the "ore").

Step 3. Each term (a sequence of up to 3 consecutive words) in the mining corpus (ore), was assigned the probability of being "related to the topic" by approximating the logistic regression on the deviation from the randomness when the values of this probability was approaching 1, specifically as following:

$$Pr(t) = 1 – \exp(-(s - 1) / a), \text{ where}$$

s = signal to noise ratio of the term, estimated as:

$$s = (df_m / N_m) / (df_w / W), \text{ where}$$

df_m was the number of occurrences of the term in the mining corpus,

N_m was the number of pages in the mining corpus,

df_w was the number pages on the Web in which the terms occurs, obtained by querying Google,

W was the total number of pages covered by Google, set to *3,000,000,000* at the time,

df_m / N_m represented "signal", while df_w / W represented the "noise." For the non related term, we would expect the proportion of the pages within the mining corpus that have this term to be the same as the proportion of the pages having this term on the entire Web. The ratio of those two proportions represented the deviation from randomness within the mining corpus.

The adjustment parameter *a* defined how "steep" the probability curve was relatively to the signal to noise ratio. We set *a* to .5 by visually inspecting the related concepts for the different topics from the preceding years. With this value, the signal to noise ratio of 1.5 would give the probability of 1 – exp (-1) = .63. The signal to noise ratio of 2.5 would result in p = .86, etc. In this application, the outcome was not sensitive to the value of parameter *a* since we only needed to select the top most deviated from the background terms and did not need the actual probability estimate. However, it would still be needed for a more general application of the approach, e.g. as in Belkin at al. (2005).

Tag	Original Query	Expansion Model	Expansion Coefficient	Geometric MAP over all topics	Best Geometric MAP of all participants	Median Geometric MAP of all participants
ASUBE	Title + Desc	Linear	.1	0.1840	0.2326	0.1256
ASUBE3	Title + Desc	Linear	.3	0.1772	0.2326	0.1256
ASU DIV	Title + Desc	Structured Query	.3	0.1400	0.2326	0.1256
ASUDE	Desc	Linear	.3	**0.1784**	**0.1784**	0.1028
ASUTI	Title	Structured Query	.3	0.0616	0.2326	0.1293

Table 4 ASU Official Robust Runs.

Table 4 summarizes our official robust runs. We used two models for expanding the query: *Linear* and *Structured Query*.

Linear model used the baseline created by BM25 retrieval model from Lemur 3.1 package with the default parameters enhanced with the pseudo relevance feedback with the parameters estimated based on Robust 2004 topics: feedbackDocCount = 20, feedbackCoefficient = .3, feedbackTermCount = 100). To expand the original query, we implemented our own module using the available C++ source in Lemur package. The top 1000 documents from the baseline were re-ranked according to the following score:

score = original score + expansion coefficient * expansion score,

where the expansion score was obtained using BM25 ranking with the default parameters for the query consisting only of the top 10 mined terms obtained after step 3 described above.

Structured Query used StructQueryEval application from Lemur 3.1 combining all the expanded terms under a single #swum operator and the specified expansion coefficient.

Overall, our results were very encouraging: our "description only" run was the best among all participants; our "title + description" runs were well above the median across all participants. Our "title only" run was below the median, however we conjecture that some groups mislabeled their "title + description" runs as "title only", which seems to be the only explanation why the best "title only" score and "title + description" scores were identical up to the all 4 digits reported.

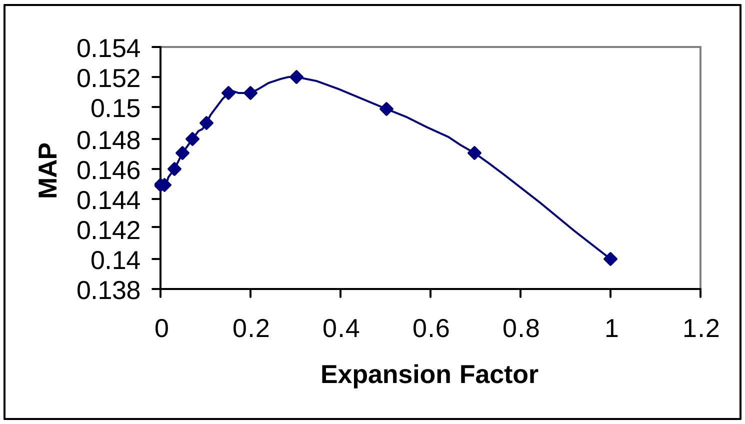


Figure 6. Improving the performance of the baseline obtained by pseudo relevance feedback using Internet-based expansion tested on 2004 collection and judgments.

Since a limited number of official runs to submit does not allow to test the importance of each of the technique and parameter involved, we run additional tests varying the expansion factor. Figure 6 illustrates how Internet-based expansion improves the baseline obtained by using BM25 combined with pseudo relevance feedback (PRF). Since this year topics were a subset of last year topics, we were able to run some additional tests with this year topics and last year judgments. The PRF parameters were optimized first, then the Internet-based expansion was applied as described above under "Linear Model." The expansion achieved additional improving even on top of PRF baseline, with the peak improvement at approximately 8%.

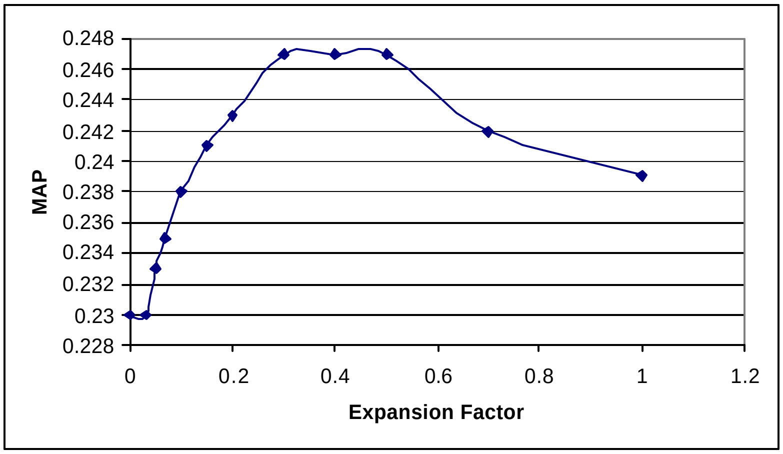


Figure 7. Improving the performance over the weak baseline by mining-based expansion (based on year 2005 judgments).

Figure 7 shows the mean average precision (MAP) as the function of the expansion factor when the mining-based expansion was applied to the weak (no PRF) baseline. The maximum value of .247 was achieved at the expansion factor of .5. The optimal improvement was statistically significant at the level of alpha <.05.

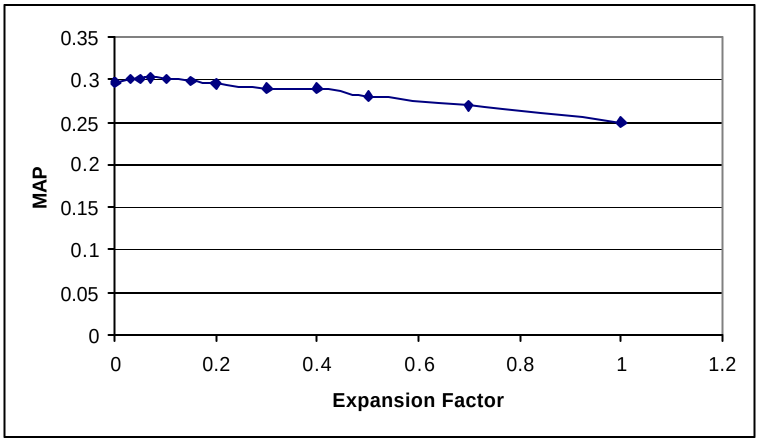


Figure 8. Improving the performance over the strong baseline by mining-based expansion (based on year 2005 judgments).

Figure 8 shows the MAP as the function of the expansion factor when the mining-based expansion was applied to the (strong) baseline. The maximum value of .298 was achieved at .2. The maximum improvement was very small and not statistically significant at the level of alpha =.1.

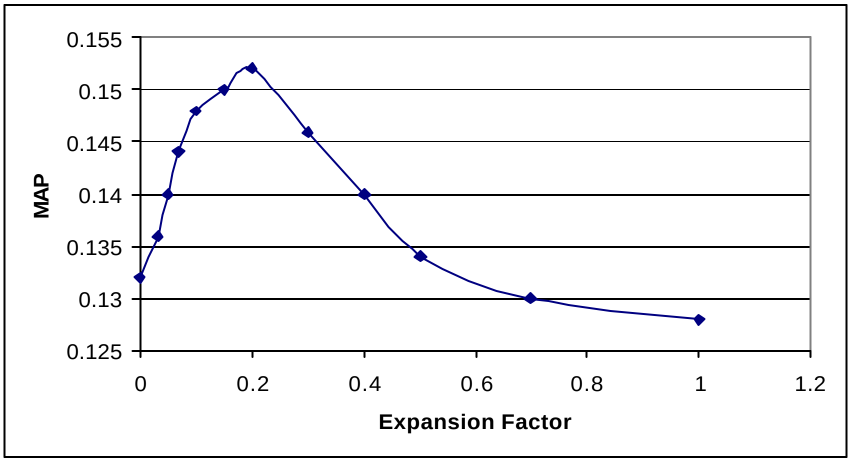


Figure 9. Improving the performance over the weak baseline by mining-based expansion. The year 2004 judgments.

For large expansion factors the effect was negative possibly due to too much drift from the original query. However, the prior research (e.g. Roussinov et al., 2005) and our experience with 2004 test sets indicated that the range up to .4 is always "safe", with the peak typically around .1-.2. Same behavior of the safe and optimal values have been noticed with the PRF technique as well. In general, determining the exact values of the optimal expansion factor in practical applications and while testing research hypothesis can be done using machine learning paradigms (e.g. Roussinov & Fan, 2005).

CONCLUSIONS AND FUTURE DIRECTIONS

From the competitive aspect, our results are encouraging: in all the (sub) categories that we entered (factoid, other, document ranking, robust) our runs are above the median performance across all participants, with QA document ranking ranked #6 and Robust runs being well over the mean. In the "description only" condition within Robust retrieval our run was the best among all participants. Our redundancy based Internet mining technology was also beyond several runs submitted by Rutgers group (Belkin et al., 2005), with also good performance. Based on those results and the algorithmic approaches described in this paper we conclude that redundancy based approaches to question answering and document retrieval are likely to be an interesting research direction, the conclusion that is well in line with the modern understanding that large amounts of training data are crucial for successful machine learning applications, possibly even more important than the choice of algorithms or investments into manually codified knowledge. Specifically, we have been able to capitalize on the vast amount of textual data available on the Web and made accessible by commercial search engines. The topical diversity of this data allowed to easily find thousands of pages about each topic or target and successfully analyze it to mine for the answers (as repeated patterns) or for the suitable expansions for the topic queries (as a set of more frequent than in background terms). At the moment, we are working on a number of extensions to the reported work, specifically: designing accurate expansion techniques based on language models, making answer patterns more "semantic," and trying the developed technology within practical applications.

ACKNOWLEDGEMENTS

This work was partially supported by Dean's Award of Excellence, W.P. Carey School of Business, Arizona State University, Summer 2005.

REFERENCES

N.J. Belkin, M. Cole, J. Gwizdka, Y.-L. Li, J.-J. Liu, G. Muresan, D. Roussinov, C.A. Smith, A. Taylor, X.-J. Yuan. Rutgers Information Interaction Lab at TREC 2005: Trying HARD. *In proceedings of TREC 2005*, Nov. 15-18, 2005.

Bikel, D., Schwartz, R., and Weischedel, R (1999). An algorithm that learns what's in a name. *Machine Learning*, 34:211–231, 1999.

Chau, M., Chen, H., and Zeng, D. (2001). Personalized Spiders for Web Search and Analysis. *Proceedings of the 1st Joint Conference on Digital Libraries*, Roanoke, Virginia, June 2001, pp. 79-87.

Chen, H., Fan, H., Chau, M. and Zeng, D. (2001). MetaSpider: Meta-searching and Categorization on the Web. *Journal of the American Society for Information and Science and Technology*, 52(13), 1134-1147.

Claburn, T., Search For Tomorrow. (2005). Information Week, March 28, 2005, http://www.informationweek.com/story/showArticle.jhtml?articleID=159905922

Cronen-Townsend, S., Zhou, Y., and Croft, W.B. (2002). Predicting query performance. *In Proceedings of the ACM Conference on Research in Information Retrieval (SIGIR)*, 2002.

Day, D., Aberdeen, J., Hirschman, L., Kozierok, R., Robinson, P., and Vilain, M. (1997). Mixed-Initiative Development of Language Processing Systems. In *Proceedings of the ANLP Conference*.

Dumais, S., Banko, M., Brill, E., Lin, J., and Ng, A. (2002). Web Question Answering: Is More Always Better? *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, Tampere, Finland, August 11-15.

Filatova, E. and Hatzivassiloglou, V. (2003). Domain-independent detection, extraction, and labeling of Atomic Events. *In Proceedings of the RANLP Conference*.

Koenemann, J., and Belkin, N. J. (1996). A case for interaction: A study of interactive information retrieval behavior and effectiveness. In Proceedings of the Human Factors in Computing Systems Conference (CHI'96). ACM Press, New York, 1996.

Kwok, K.L., Grunfeld, L., Sun, H.L., Deng, P. and Dinstl, N. (2004). TREC2004 Robust Track Experiments using PIRCS. *In D. K. Harman, editor, Proceedings of the Twelve Text Retrieval Conference, NIST Special Publication*, 2003.

Lin, J. (2005). Evaluation of Resources for Question Answering Evaluation. *Proceedings of ACM Conference on Research and development in information retrieval*, 2005.

Roussinov, D. and Robles, J., Ding, Y. (2004). Experiments with Web QA System and TREC2004 Questions. In the proceedings of TREC conference. November 16-19, 2004, Gaithersburg, MD.

Roussinov, D., and Fan, W. (2005). Discretization Based Learning Approach to Information Retrieval. In *proceedings of 2005 Conference on Human Language Technologies*.

Roussinov, D., and Zhao, L. (2003). Automatic Discovery of Similarity Relationships through Web Mining, *Decision Support Systems*, 35, 2003, pp. 149-166.

Roussinov, D., Zhao, L., and Fan. W. (2005). Mining Context Specific Similarity Relationships Using The World Wide Web. *In proceedings of 2005 Conference on Human Language Technologies.*

Selberg, E. and Etzioni, O. (1995). Multi-Service Search and Comparison using the MetaCrawler. *Proceedings of the 4th World Wide Web Conference*, Boston, MA, USA, December 1995.

Vogt, C., Cottrell, G., Fusion Via a Linear Combination of Scores. *Information Retrieval*, 1(3), pp. 151-173.

Voorhees, E. and Buckland, L.P., Eds. (2003*). Proceedings of the Twelve Text Retrieval Conference TREC 2003*.

Voorhees, E. and Buckland, L.P., Eds. (2004*). Proceedings of the Thirteenth Text Retrieval Conference TREC 2004*.