Exploiting Diversity for Answering Questions

John Burger and **John Henderson**
The MITRE Corporation
202 Burlington Road
Bedford, MA 02144, USA
{john,jhndrsn}@mitre.org

Abstract

We describe initial experiments in combining the output of question answering systems using data from the 2002 TREC Question Answering task. We explore several distance-based combining methods, as well as a number of distance metrics involving both word and character ngrams.

1 Introduction

Progress in question answering technology can be measured as individual systems improve in accuracy, but it is not the only way to witness technological progress. A question one can ask is how well we can perform automatic question answering as a community. If we were asked to enter an Earth English system in an intergalactic TREC, how well would we do? One easy answer is that we would perform as well as the best QA system. A second answer is that perhaps we could do even better by combining systems—this might be expected to work if different systems were independent in their errors. The follow-up question is how would we build such a system?

Lower bounds on the highest possible performance current technology can achieve on a given dataset have practical value, as well. They allow us to better estimate how well systems are doing with respect to the underlying difficulty of the dataset, and continually provide performance targets that are known to be achievable. Without such lower bounds on optimal performance, one cannot determine if technological progress in a domain has simply stalled.

NIST's ROVER system for combining speech recognizer output gives ASR researchers an updated goal to shoot for after every evaluation, as well as an implicit measure of the extent to which systems are making the same errors (Fiscus, 1997). The work herein initiates a similar set of experiments for question answering technology.

2 Background: The TREC Question Answering Task

Under the auspices of the National Institute for Standards and Technology (NIST), the Text Retrieval Conferences (TREC) have been an annual opportunity for the information retrieval community to evaluate techniques in a variety of tasks. For the last four years, TREC has included a *question answering* activity, wherein commercial and academic groups from around the world can evaluate systems designed to retrieve answers to questions, rather than simply documents from queries (Voorhees, 2002). This last year, 34 groups participated by running their systems on 500 previously unseen questions, against a corpus of approximately one million newswire documents. The task was to retrieve a single, short phrasal answer to each question from one of the documents, returning the answer string along with the document identifier. Answers were evaluated as strictly correct by the NIST assessors only if the indicated document justified the answer appropriately, and if no extraneous material was included in the answer string.

For example, Question 1399 from the 2002 evaluation was:

> *What mythical Scottish town appears for one day every 100 years?*

Participating systems returned *Hong Kong*, *Tartan*, *Lockerbie*, and *Brigadoon*, as well as a number of other candidates. Only *Brigadoon* was judged to be correct, and only if the system also pointed to a document that explicitly justified that answer—a document that simply mentioned the town was insufficient. Systems also had the option of indicating that they believed a question to be unanswerable from the corpus, by returning the NIL document ID.

This year, TREC QA participants were encouraged to develop confidence assessment techniques for their systems. Systems returned the answer set sorted by decreasing confidence that each answer was correct. This ranking was taken into account by the main evaluation metric, *average precision*. This is defined as follows:

$$\begin{aligned} \text{avg.prec.} &= \sum_{i=1}^{n} \text{prec}(i) \\ \text{prec}(i) &= \frac{\text{\# correct answers up to rank } i}{i} \end{aligned}$$

where n is the total number of answers in the evaluation set. In this way, correct answers near the top of the system's ranking count for far more than those near the bottom, and systems are rewarded for good confidence estimates.

3 Methods

The task we are faced with is straightforward. Given a collection of answers to a question, choose the one most likely to be correct. For our purposes, each answer consists of the answer string and an identifier for an associated document. Our data was initially limited in that it did not indicate which answers were provided by which system—see the discussion below. Note that we use no knowledge of the question or of the document collection. Our assumption is that the authors of the individual systems have milked the information in their inputs to the best of their capabilities. Our goal is to combine their outputs, not to re-investigate the original problem.

In TREC 2002's main QA evaluation there were 67 different systems or variants thereof involved. Thus, our corpus consists of 67×500 answers. To guard against any implicit bias due to repeated experimentation on the small dataset available, we randomly selected a 100-question subset for development of our techniques—the remaining 400 questions were kept as a test set, evaluated only once, when development was complete. While we may have wished to pursue parametric techniques, we felt that this training set was too small to explore any but the simplest (non-parametric) techniques. An exception is the experiments described below involving priors over the document sources and the systems themselves.

Voting is an easily understood technique for selecting an answer from among the 67 suggestions. Unfortunately, voting techniques do not provide a mechanism for utilizing full knowledge of partial matches between proposed answers. While his original goal was the selection of representative DNA sequences, Gusfield (1993) introduced a general method for selecting a candidate sequence that is close to an ideal centroid of a set of sequences. His technique works for all distance measures that support a triangle inequality, and offers a bound that the sum of pairwise distances (SOP) from proposed answers to the chosen answer will be no more than twice the SOP to the actual centroid (even though the centroid may not be in the set). This basic technique has been used successfully for combining parsers (Henderson, 1999). Appealingly, the centroid method reduces to simple voting when an "exact match" distance is used (the complement of the Kronecker delta).

One advantage of both simple voting and the centroid method is that they give values (distances) that are comparable between questions. An answer that receives 20 votes is more reliable than an answer that receives 10 votes, and likewise for generalized SOP values. This gives a principled method for ranking results by confidence and measuring average precision, as required for this year's TREC evaluations.

In selecting appropriate distance measures between answers, both words and characters were explored as atomic units of similarity. Two well-known non-parametric distances are available in the literature: Levenshtein edit distance on strings and Tanimoto distance on sets (Duda et al., 2001). The latter is defined as follows:

$$\mathrm{D_T}(S_1, S_2) \quad = \quad 1 - |S_1 \cap S_2| / |S_1 \cup S_2|$$

We experimented with each of these, and also generalized the Tanimoto distance to handle multisets by defining the obvious function to map multisets to simple sets: Given a multiset containing instances of a repeated element Fx we can create a simple set by subscripting, e.g., $\langle x, x, y, z \rangle \rightarrow \{x_1, x_2, y, z\}$. We can then use the standard Tanimoto distance on the resulting simple sets.

Overall, systems seemed to be conservative and answered with the NIL document (no answer) at a rather high rate (17% of all answer strings this year). To compensate for this, a "source prior" was collected from the 100-question training set. These four numbers recorded the accuracy expected when systems generated answers from the four document sources (Associated Press, New York Times, Xinhua News, and NIL). Those numbers were then used to scale the distance measures for the corresponding answer strings. Other than these priors, no other features of the document ID string were used.

4 Experiments and Results

Several measurements were made to ascertain the quality of the various selection techniques, as seen in Figure 1. Precision, P, indicates the accuracy of the technique, the percentage of the answers that were judged to be correct. avgP is the main measure used by NIST this year—the average precision of all prefixes of the sequence of answers placed in order of high to low confidence. Strict corresponds to the correctness criterion used by NIST—the answer must be exact and justified by the referenced document (assessor judgment $= 1$). The Loose figures discard these two criteria (assessor judgment > 1). The Loose P measure was the one that was optimized during development.

In Figure 1 we see both development and test set results for answer selection experiments involving a sample of the distance measures with which we experimented, as well as the best-performing system involved in the evaluation. All of the design and selection of the distance measures was done using hill-climbing on the development set, and only after this exploration was complete was the performance on the test set measured. Two general observations can be made about these results (and others not shown): taking into account a prior based on the document source (including NIL) is useful, as is working with

	Dev Set (100 Qs)				Test Set (400 Qs)	
	Strict		Loose		Strict	
	P	avgP	P	avgP	P	avgP
exact string match	50	70	54	74	42	65
word set	54	75	58	78	46	68
word bag	54	75	58	78	46	68
character set	51	65	57	67	46	62
character bag	60	81	64	85	50	74
word bag w/ doc priors	66	83	74	88	51	72
character bag w/ doc priors	64	81	69	86	50	72
5-character bag w/ doc priors, weighted numeric strings	66	85	76	90	53	73
Best TREC system	84	82	86	90	83	86

Figure 1: Answer selection results (percentages, best results in bold)

feature bags from the answers rather than sets. The best-performing selection system used all character strings of length 5 and less as features, combined with the multiset Tanimoto distance measure described above, and scaled with document source priors. Furthermore, a numeric string mismatch was weighted to be twice as costly as mismatching a non-numeric string.

Question 1674 provides an example that contrasts this best selector with a simple voting scheme (exact string match):

What day did Neil Armstrong land on the moon?
1969 (simple voting—incorrect)
July 20, 1969 (best measure above—correct)

While a plurality of systems answered with *1969*, many others answered with variants of the correct answer that differed in punctuation, as well as *on July 20, 1969*; *July 18, 1969*; *July 14, 1999*; even simply *20*. All of these, including the incorrect instances of *1969*, contributed to the correct answer being selected.

5 Discussion and Conclusions

The disparity between the dynamic range of these techniques on the development and test datasets suggests that the dev set sample size of 100 (6700 proposed answers and NILs) may be too small to draw conclusions on the relative quality of selection techniques. Still, consistencies in rank orderings of selection techniques between the two datasets suggest that these methods of system combination are effective.

None of our combinators did as well as the best TREC system on the test dataset. It is important to note that in these experiments we did not have access to several useful evidence sources. First, this year's submissions included system estimates on answer confidence, if only implicitly. The selection mechanism could take advantage of this by weighting each submitted answer string appropriately. Second, past TRECs show that some systems are reliably more accurate than others, and if each answer string were labeled with a system ID, even if anonymized, we could use system-level features in the selector, such as a simple prior. Given sufficient training, we might even take question features into account, learning that certain systems are better at certain types of questions. We would like to pursue the use of these and other evidence sources in the future.

References

Richard O. Duda, Peter E. Hart, and David G. Stork. 2001. *Pattern classification*. New York: John Wiley & Sons.

Jonathan G. Fiscus. 1997. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER). In *Proceedings of the European Conference on Speech Technology*, volume 4, pages 1895–1898.

Daniel Gusfield. 1993. Efficient methods for multiple sequence alignment with guaranteed error bounds. *Bulletin of Mathematical Biology*, 55(1):141–154.

John C. Henderson. 1999. *Exploiting Diversity in Natural Language Processing*. Ph.D. thesis, Johns Hopkins University.

Ellen M. Voorhees. 2002. Overview of the TREC 2001 question answering track. In *The Tenth Text REtrieval Conference (TREC 2001)*, pages 42–51. NIST Special Publication 500-250; available at http://trec.nist.gov/pubs/trec10/t10_proceedings.html.