Answer Attenutation in Question Answering

Katie Bell and **James R. Curran**
School of Information Technology
University of Sydney
Sydney, Australia
{kbel5892, james}@it.usyd.edu.au

Abstract

Research in Question Answering (QA) has been dominated by the TREC methodology of black-box system evaluation. This makes it difficult to evaluate the effectiveness of individual components and requires human involvement. We have collected a set of answer locations within the AQUAINT corpus for a sample of TREC questions, in doing so we also analyse the ability of humans to retrieve answers. Our answer corpus allows us to track answer attenuation through a QA system. We use this method to evaluate the Pronto QA system (Bos et al., 2007).

1 Introduction

A Question Answering system, is any system which answers questions posed in natural language. In its earliest forms, QA systems were natural language front-ends to structured databases of knowledge (Androutsopoulos, 1995). Today, there exists a massive quantity of data freely available on the internet in raw textual form, but to find specific information the user may be required to read many documents returned from a query. The goal of open domain QA is to enable users to find specific answers to questions from enormous corpora spanning many domains. QA can be seen as a search problem: finding answers in a corpus of text. A QA system reduces the search space in stages, starting with selecting documents, then passages, and so on, until a single answer is returned.

The current method of evaluating Question Answering systems stems from the Text REtrieval Conference (TREC) Question Answering track. Using a standard document collection and question set, each participating system is graded on the proportion of answers which are correct and supported by the source document. These criteria are assessed by the only reliable way to determine the correctness and support of an answer: humans. However, manual evaluation is costly and not feasible for constant evaluation of QA while developing new systems or techniques. For this purpose a set of correct answer regular expressions are crafted from the correct answers as judged by the TREC QA human assessors. These answer keys are unreliable as they are both not exhaustive and do not take into account the support of the source document (Lin, 2005).

For reliable automated evaluation we need a gold standard dataset. This would be an exhaustive list of the exact locations of human verified correct answer instances for a set of questions. Using such a set of answer locations, we can define an extension of current evaluation practices, which we call *answer attenuation*. Answer attenuation is the proportion of correct answers lost from the search space after each stage of processing. The purpose of this measure is both to provide more information about the effectiveness of each system component and a more reliable overall system evaluation.

With group of volunteer annotators, we have collected an initial test set of answer locations within a document collection which is an approximation of this theoretical gold standard. From analysing the annotators' behaviour we find that on an individual scale, humans are not adept at exhaustively finding answers. We used this data to analyse the answer attenuation of the Pronto QA system (Bos et al., 2007). In doing so, we revealed some weaknesses both in Pronto and in the use of TREC answer keys as an evaluation measure.

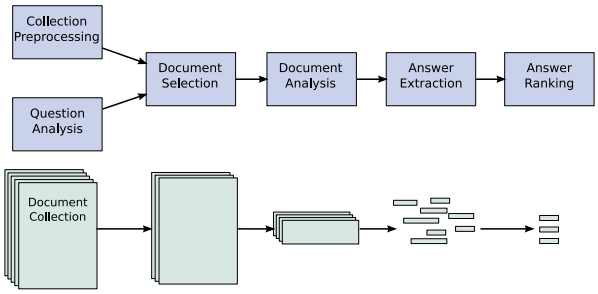

Figure 1: The generic structure of QA systems

2 Background

Question answering can be seen as a search problem. QA systems typically follow a pattern of successively narrowing the search space for answers. The generic structure defined by Hirschman and Gaizauskas (2001) provides a model for describing QA systems. Not all QA systems include all of these components, yet it provides a useful framework to draw parallels. The generic system structure is shown in Figure 1, starting with pre-processing of the document collection, usually indexing to improve retrieval time. The question analysis stage involves determining the expected answer type of the question and generating keywords to be used in retrieving documents in the document selection stage. The document analysis stage involves selecting sections from the text which are considered likely to contain answers. Short phrases which match the expect answer type of the question are selected in the answer extraction stage, these answers are then ranked according to how well they match or answer the original question.

2.1 QA Evaluation

The methods for evaluating and analysing QA systems can be divided into three categories: black-box whole system evaluation, component specific evaluation and system-wide component analysis.

Black-box Evaluation The results of a human assessment of an answer can be approximated using human generated answer keys (Breck et al., 2000). This method remains the most widely used form of evaluation of QA systems. However, this method should be used with caution (Lin, 2005). Lin's work calls into question the reliability of the use of these answer keys for system evaluation and comparison. Firstly, the answer keys do not take into account the support of the source document. Similarly, the answers are limited to those returned by existing systems. Any new QA system will not be evaluated as better than the sum of preceding systems. Consequently, Lin finds that these evaluation resources underestimate answer accuracy.

Lin proposes a potential solution as future work which is very similar to the method proposed here: the tagging of all instances of the correct answer for each question. He then highlights the difficulties of such an approach, particularly in the creation of the test data. There is no efficient way to exhaustively find all answer instances without a manual search of the entire document collection. Searching for the known answer string is unreliable as answers can appear in multiple forms, for example a date can appear as last Monday. We can never guarantee a completely exhaustive corpus. However, current QA systems have no reached performance levels where an exhaustive corpus is necessary to measure improvement. We use the results of our human annotations as an approximation to this ideal corpus. Lin only described the usefulness of this method as a replacement for the current black-box system evaluation but did not consider the additional use of this method for component analysis of QA systems.

Component Specific Evaluation Several groups have realised the importance of evaluating individual components independently of the whole system and have attempted to do so for specific components, for example, answer extraction (Light et al., 2001). Particular focus has been placed on the document retrieval component. For example, substituting different algorithms then comparing overall performance (Tellex et al., 2003). Also, human assessment of QA document retrieval has formed an additional part of the TREC QA Track (Voorhees, 2003) and the results are then used for automated evaluation (Monz, 2003). These individual analyses are useful, yet their application is limited to specific components and implementations. Our approach is not component specific.

Whole System Component Analysis There are two approaches to whole system component analysis. The first approach is assessing each component's usefulness or contribution to overall system performance. This can be done by ablation experi-

ments (Brill et al., 2002), where each component is replaced by a baseline component which performs the same function in a minimal way.

The second approach is to look at cases where the system does not perform correctly, and identify the components which are causing these failures. Moldovan et al. (2003) manually traced each incorrectly answered question and decided which component was the cause. Each component was given a score in terms of the percentage of failures that it caused. This was used to determine which components of the system should be focused on to improve overall system performance. While useful, this analysis is time consuming and it is difficult to assign errors to specific components.

3 Collecting the Corpus

As the basis for the initial data collection we used the AQUAINT-1 document collection, consisting of approximately 1 million newswire articles, and a set of questions taken from the 1999 and 2000 TREC QA tracks. Also provided by NIST were expected answers and sets of regular expression answer keys intended for automatic marking.

Collecting an exhaustive list of correct answer instances is non trivial. We start with the assumption that humans are adept at finding specific information, in this case, the answers to questions. We organised a group of 20 volunteers to each spend 5 hours finding the answers to a set of TREC questions and annotating them. Reading through the entire document collection is infeasible. Thus, a web-based user interface was developed to assist the process, providing the annotators with a searching tool to first select a set of likely documents, then read them to find answers. All of the volunteer annotators were comfortable with keyword searching, primarily from the use of search engines such as Google.

Each annotator was allocated a question and given the expected answer and answers keys from TREC. They would then select some search keywords, and refine them until a manageable number of documents were returned. They would read through the documents and highlight any instances of the correct answer. Then they would create another search query and repeat the process until they were confident that they had found all of the answers.

The guidelines given to the annotators were as follows: The answer can exist in multiple forms, e.g. 1991, 25th March 1991, today. These should be tagged no matter what form or precision it is. Additionally, the whole answer should be tagged, if the document says 25th March 1991, then that whole phrase should be tagged. The annotators were also instructed that a correct answer must be supported by the sentence in which it is found, such that if a person read that sentence they would know, for certain, the answer to the question. This differs from the TREC evaluation process, which uses the entire document as support. The choice to allow only sentences both simplifies the annotation process and provides a closer correlation with how current QA systems select answers.

The questions were chosen based on the number of answers present in the document collection. This is impossible to determine without searching for, and verifying each answer. Therefore, the volunteer annotators also selected which questions would be included in the final set. Questions with fewer answer instances are most interesting for evaluation because systems are more likely to answer them incorrectly. For some questions, the number of correct answers in the document collection is overwhelmingly large and would be impossible for annotators to find all of them within a reasonable period. For these reasons, the annotators were instructed to discard questions when it became obvious that there would be more than 100 answer instances.

4 Behaviour of the Annotators

The use of human annotators was based on the assumption that humans, particularly those used to searching the internet for information, would be good at finding answers to questions. Looking at the behaviour of our volunteers, this is not the case. Humans are good at judging relevance but not at retrieval with high recall.

We analyse the shortcomings of the pilot corpus and discuss the necessary steps to enable efficient and reliable annotations of a more complete corpus.

4.1 Search Stopping Criteria

The annotators were specifically asked to find all instances of the answers for each question, however

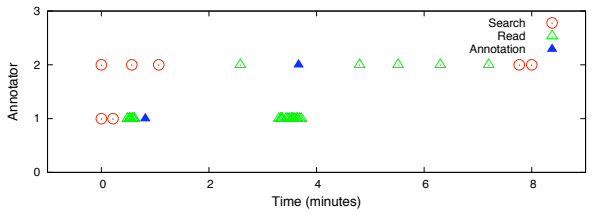

Figure 2: Time lines for What movie did Madilyn Kahn star in with Gene Wilder?

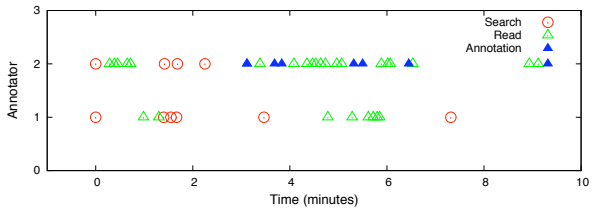

Figure 3: Timelines for Where is Tufts University?

there is no definite way to know when there are no more answers to be found. We found that the annotators used three different stopping criteria. (1) They had read all the documents returned from their searches and they believed that the search was general enough; (2) Further searches returned no new results; (3) After initially finding answers, if they read several documents with no answers.

We have plotted timelines showing when annotators searched for, read or annotated documents. Figure 2 shows two different stopping criteria. Annotator 1 stopped when there were no more documents in the search results and further searches yielded no results. Annotator 2 used a more general set of search terms and stopped annotating when they had read several documents without finding any answers. Both annotators found the same single answer.

This was not always the case. Figure 3 shows the behaviour of the two annotators assigned the question Where is Tufts University? Annotator 1, after several attempts at searching, decided that there were no answers for this question. Annotator 2 used more sophisticated queries, eventually finding several answers. From this variation in query generation ability, we know that not all of the answers have been found for each question.

Selecting the correct stopping criteria is not a trivial task and none of the criteria used by the annotators is sufficient in all circumstances.

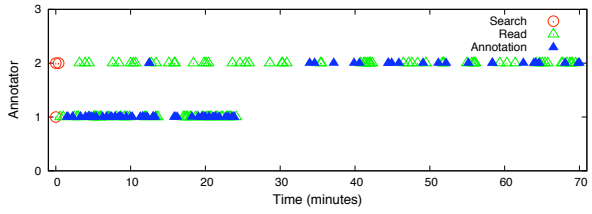

Figure 4: Time lines for Where is Venezuela?

4.2 Efficiency and Accuracy

The efficiency of the annotators related primarily to their choice of search terms. For example in the question Where is Venezuela?, shown in Figure 4. One annotator, not only was slower in reading and annotating documents, but also had fewer annotations within those documents because of the search terms used, they were less specific. The more efficient annotator searched for Venezuela AND 'south america' whereas the other annotator used keywords instead of a string literal and consequently had a higher proportion of irrelevant documents.

4.3 Search Terms

From a detailed analysis of the terms that the annotators used to retrieve documents we can both assess whether the annotators were thorough in finding answers exhaustively, and determine the annotators techniques for generating search queries.

For each question, the annotators were given the question, the expected answer and the set of answer keys. As shown in Table 1, the search terms used are usually generated from a combination of keywords from the question and from the expected answer. Unfamiliarity with regular expressions made using answer keys hard for the annotators.

The results in Table 1 were calculated based on simple case-insensitive string matching. Since the words were not stemmed, the search terms for the question, What did brontosauruses eat? had little overlap with the question because the annotators used the search term brontosaurus in combination with other words such as eat and food. The overall results suggest that the annotators used very few search terms which were not directly drawn from the question. In this example, the only word which is not a variation from the question is food.

Providing the answers for each question to the an-

Question	Overlap	Exp	Keys	Quest	None
A corgi is a kind of what?	**100**	50	50	50	0
What is sake?	50	**68**	37	25	0
What was the ball game of ancient Mayans called?	16	14	**100**	42	0
What's the average salary of a professional baseball player?	75	0	0	**100**	0
What did brontosauruses eat?	33	20	0	20	**80**
Averages	42	22	19	60	16

Table 1: The overlap of annotators search terms and the percentage source of the search terms, from the expected answer, answer keys and the question. The sample questions were chosen as the maximum of each column.

Text with annotation	#annotes	Correct
For instance , a 1993 fire that killed 188 workers at a toy factory in Thailand was similar to a **1911** New York clothing factory fire which previously held the record for the highest number of fatalities with 146 dead .	1	Incorrect
Some of the nation 's most stringent workplace-safety laws were prompted by the **1911** Triangle Shirtwaist Co. fire in New York City .	20	Correct
The decline has been particularly pronounced in high-risk industries such as mining , where the average death rate was 329 per 100,000 from **1911** to 1915 , compared with 25 per 100,000 in 1996-97 , the CDC said .	2	Incorrect
Here are some highlights : TRIANGLE FACTORY FIRE : On **March 25 , 1911** , a blaze at the Triangle Shirtwaist Co. in Manhattan 's garment district claimed 146 lives .	14	Correct
Here are some highlights : TRIANGLE FACTORY FIRE : On March 25 , **1911** , a blaze at the Triangle Shirtwaist Co. in Manhattan 's garment district claimed 146 lives .	4	

Table 2: A sample of the answer locations for the common question When was the Triangle Shirtwaist fire?

notators has further ramifications. In many cases, this lead to the annotators restricting their searches to only answers of a particular form. For example for the question When was Microsoft established? both annotators made the same search. Both annotators searched for Microsoft AND established AND 1975. Despite this high agreement, it is possible that further answers were completely disregarded, for example there could be answers of the form Microsoft was established 25 years ago today.

A total of 73% of the searches generated by the annotators contained either the expected answer or answer key in some form. The risk of making the queries answer specific is that answers in other formats will be missed. If the expected answer and answer keys were not provided, the annotators would be more likely to search for answers of any form. However, this has the disadvantage that the annotators would be less efficient in finding the answers.

The overall behaviour of the annotators focuses on speed and efficiency rather than accuracy or thoroughness, as the annotations were done within a short amount of time, with the goal of getting as much data as possible. With more time allowed for the annotations, more detailed guidelines and training, a far more exhaustive set of answer instances could be found for the question set.

4.4 Reliability of Manual Annotations

In order to study how accurately and exhaustively the annotators were finding the answers, all annotators were required to answer a common question: When was the Triangle Shirtwaist fire? Some results from this common question are shown in Table 2. These results show that not all of the answers which were marked were supported by the text surrounding them. Some annotators did not always follow the instruction that as much of the answer should be annotated as possible. However, these annotations are still considered correct because they overlap. With overlapping annotations the longest answer is used in the final answer set. Upon manual inspection, the majority of annotations for this question were correct, the highest agreement for an incorrect (unsup-

ported) answer was 9, and the lowest agreement for a correct answer was 16 out of the 20 annotators. This is a significant gap and indicates that when there are enough annotators using only the answers which the majority of annotators agree upon is reliable both in terms of coverage and accuracy.

In addition to the common question, half of the set of questions were annotated by at least two annotators. A sample of questions and the overall annotation agreement results are shown in Table 3. The question Who was President Cleveland's wife? is an example of where both annotators failed to find all of the answers, both annotators found a different correct answer to the question. It is likely that there are more answers which were not found by either annotator. There were also clear mistakes in the dataset such as the question What is a nematode?, the two annotators assigned this question had annotated almost exactly the same sentences, however one of the annotators had mistakenly tagged the word nematode in each sentence and not the actual answer to the question.

In using this annotated answer set, there are two options, the first is to only use annotations which both annotators have agreed upon, this will significantly reduce the erroneous answers, however it would also reduce the coverage of the answer set. When using the answer set to analyse a QA system, the system would appear to be performing worse than it actually is. The other option is to include all answers regardless of whether multiple annotators agreed on them. This would result in the system appearing worse than it actually is when it does not retain the wrong answer, however it will appear to be performing better than it actually is if it retains the wrong answers. We chose to include all of the answers to maximise coverage, as this it is coverage which is one of the faults of the current standard evaluation system. Ideally, there would be more annotators for each question, and thus the inclusion of each answer location could be decided with relative confidence by a majority vote, as shown in result of the common question described above.

5 Using Answer Attenuation

When constructing or modifying a QA system, there is a trade-off between speed, memory and how much of the document collection is retained at each stage. For example, if at the document selection stage, all of the documents are selected, this guarantees that all answers in the document collection are still within the search space, however the following stages will take much more time to process this larger search space. Alternatively only one document or passage could be selected at each stage. This would result in very fast processing in the later stages however most of the answers will be lost and assuming each stage involves some probability of error, it is possible that all correct answers are lost resulting in the system returning the wrong answer or no answer. Consequently a QA system designer must balance these two concerns: efficiency and recall. This is where our answer attenuation measure is critical.

Using the answer set we collected, it is possible to measure the proportion of this answer set which is still in the QA system's search space after each stage of processing each question. What defines a stage of processing, is any work done by the system which narrows the search space for answers, whether this is intentional or not. For example, document retrieval is intentionally narrowing the search to a specific set of documents, whereas a parsing step removes sentences if they fail to parse, an effect which is unintentional. Consequently, answer attenuation serves the dual purpose of error detection as well as an analysis tool to fine tune the QA system to maximise performance.

6 Results

We compared answer attenuation to the TREC answer keys in two ways. Firstly, we compared all of the annotated answers to check whether they would have been marked as correct by the TREC answer keys. Secondly, we calculated the answer attenuation for all six stages of the Pronto QA system. The evaluation of the final answer, we compared with the TREC answer key evaluation. Our final evaluation of answer attenuation is to show that it is useful in detecting and repairing flaws in QA systems.

6.1 Comparing Annotations with Answer Keys

The final answers produced by Pronto were marked by both the TREC answer keys and whether the final

Question	Annotations	Answers	Aggreement
Who was President Cleveland's wife?	2	2	0%
When was Microsoft established?	2	1	100%
What is a nematode?	16	16	0%
Where does chocolate come from?	31	23	34%
Where is Trinidad?	22	21	4%
Who was the 33rd president of the United States?	10	5	100%
What province is Edmonton located in?	34	28	21%
Average for all questions	17.0	13.9	31.2%

Table 3: A sample of questions and the agreement when a question was answered by two annotators

answer location was one of those in the answer set. A difference between these two scores must have one of three possible causes: (1) there is a correct answer in the document collection which the annotators did not find; (2) the answer key marks the answer as correct when it is not supported by the document; (3) the answer returned was annotated incorrectly by the annotators. In all of the experiments run, this third case never occurred, despite Pronto answering most of the questions wrong, and the existence of several incorrect answers in the answer set. The question What is the busiest air travel season? is an example of the TREC answer key marking as correct when the document does not actually answer the question. The source document refers to a specific summer in which there were many air traffic delays, not that the season of summer is the busiest time to travel or even busy at all. Similarly for the question Where is Tufts University?, the answer Boston is correct according to the TREC answer keys but the document from which it was taken does not support it.

That document was read by both annotators who were assigned that question yet was not annotated, indicating their belief that it did not answer the question. In every case where the returned answer was not one of the annotations and the TREC answer key marked it as correct, the document was found by a human assessment to not support the answer. This drawback in the answer key method is clearly shown by marking all of the human annotations according to the TREC answer keys, the results of this comparison is shown in Table 4. Only 42% of the answers found by the annotators would be considered correct according to an evaluation based solely on the answer key. Additionally for some of the questions, none of the answers marked by the annotators would have been marked as correct by the TREC answer key.

7 Answer Attenuation in Pronto

The Pronto QA system has 6 discrete components which actively reduce the search space. The document retrieval component selects the document identifiers from an index based on search terms, these documents are extracted into a single file which is parsed with the wide-coverage CCG Parser (Clark and Curran, 2004). This syntactic analysis is then used by a component called Boxer to generate a semantic analysis of the text (Bos, 2005). The document retrieval and extraction components form the document selection stage according to the generic QA system structure. Similarly, the CCG and Boxer stages form the document analysis stage. The matching component performs answer extraction, and 'select' refers to the answer ranking stage. Pronto is currently in a state of development and is consequently not performing to the standard expected of a state of the art system.

Component	Q	%L	Run1	%L	Run2	%L
Total Ans.	10	0	872	0	872	0
DocRet	4	60	493	43	493	43
Extract	4	0	418	15	493	0
CCG	4	0	345	17	415	15
Boxer	4	0	345	0	415	0
Matcher	1	75	30	91	27	93
Select	1	0	5	83	4	85
Answer	1	0	3	40	3	25

Table 5: The answer attenuation for Pronto. Showing the answers remaining in the search space and the % loss for an example question (Q) and two runs.

We measured the number of correct answers in the search space at the beginning and end of each

Statistic	Min	Max	Total	Avg. per Question
Number of annotations	0	339	1532	14.7
Number of annotations which matched answer keys	0	336	833	8.0
Percent of annotations correct according to answer keys	0%	100%	54.4%	41.6%
Number of answers annotated	0	69	1058	10.2
Number of answers which matched answer keys	0	44	446	4.3
Percent correct according to answer keys	0%	100%	42.2%	41.1%

Table 4: For each question, the number of answers the annotators found and the percentage of these which would be judged correct by the answer keys

stage. An example question, What is the wingspan of a condor? is shown in Table 5, there were a total of 10 answers in the annotated answer set for this question. Only 4 answers were found in the document retrieval stage, and 3 of those answers were lost in the matching stage. The system returned the remaining correct answer as its final result. The first run shows the initial results aggregated over all question. We expected the document extraction phase to have zero loss. This stage only copies the documents from the collection into a single file and is not intended to reduce the search space. This was traced to a problem in the way Pronto handles results form multiple searches and the problem repaired for the second run. This repair improved the individual component score, yet did not improve the overall system score, hence by black box evaluation the repair would not be detectable.

8 The Ultimate Answer Set

Our results indicate that with a larger group of annotators, a more complete set of answers could be found. We propose the creation of a collaborative dataset, emulating the success of open projects such as Wikipedia. Using a fixed document collection and question set, an initial set of answer locations is used as an evaluation set alongside the TREC answer keys and manual evaluations. Whenever a new answer is found, it is added to the dataset if a threshold number of humans agree on the answer. Similarly, if an answer is included in the collection which is agreed to be incorrect or not have enough support from the source document, it is removed. Over time the answer set is refined to be both exhaustive and accurate.

9 Conclusion

The answer attenuation measure provides both a useful analysis of QA system components and has the potential to also be a reliable whole system performance evaluation. This depends on the creation of a sufficiently exhaustive corpus of answer locations. We have created a pilot corpus of answer locations, and by studying the behaviour of the annotators we conclude that it is not yet reliable enough to replace current methods of overall evaluation. What was lacking in QA evaluation was an automated method for conducting an evaluation across all system components.

The next step is the creation of a larger high quality answer location corpus. This would involve more detailed guidelines for the annotators, more annotators assigned to each question and more time allowed. The resulting corpus could then be improved collaboratively over time as it is used to evaluate QA systems.

The creation of our pilot corpus has shown that this goal is feasible. We have used this corpus to analyse answer attenuation in Pronto, and have shown that it can reveal flaws in existing question answering systems.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful feedback, Johan Bos who helped with our installation of Pronto, and our 20 volunteer annotators for helping us test these ideas.

References

Ion Androutsopoulos. 1995. Natural language interfaces to databases - an introduction. *Journal of Natural Language Engineering*, 1:29–81.

Johan Bos, James R. Curran, and Edoardo Guzzetti. 2007. The Pronto QA System at TREC 2007: Harvesting Hyponyms, Using Nominalisation Patterns, and Computing Answer Cardinality. In *TREC 2007*.

Johan Bos. 2005. Towards wide-coverage semantic interpretation. In *Proceedings of Sixth International Workshop on Computational Semantics IWCS-6*, pages 42–53.

Eric J. Breck, John D. Burger, Lisa Ferro, Lynette Hirschman, David House, Marc Light, and Inderjeet Mani. 2000. How to Evaluate your Question Answering System Every Day and Still Get Real Work Done. In *LREC-2000*, pages 1495–1500.

Eric Brill, Susan Dumais, and Michele Banko. 2002. An analysis of the askmsr question-answering system. In *EMNLP '02*, pages 257–264, Morristown, NJ, USA. Association for Computational Linguistics.

Stephen Clark and James R. Curran. 2004. Parsing the WSJ using CCG and Log-Linear Models. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL '04)*.

L. Hirschman and R. Gaizauskas. 2001. Natural language question answering: the view from here. *Nat. Lang. Eng.*, 7(4):275–300.

Marc Light, Gideon S Mann, Ellen Riloff, and Eric Breck. 2001. Analyses for elucidating current question answering technology. *Natural Language engineering*, 7(4):325–342.

Jimmy Lin. 2005. Evaluation of resources for question answering evaluation. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 392–399, New York, NY, USA. ACM.

Dan Moldovan, Marius Paşca, Sanda Harabagiu, and Mihai Surdeanu. 2003. Performance issues and error analysis in an open-domain question answering system. *ACM Trans. Inf. Syst.*, 21(2):133–154, April.

Christof Monz, 2003. *Document Retrieval in the Context of Question Answering*, chapter 8, page 546. Springer Berlin / Heidelberg.

Stefanie Tellex, Boris Katz, Jimmy Lin, Aaron Fernandes, and Gregory Marton. 2003. Quantitative evaluation of passage retrieval algorithms for question answering. In *SIGIR '03*, pages 41–47, NY, USA. ACM.

Ellen M. Voorhees. 2003. Overview of the trec 2002 question answering track. In *Proceedings of the Eleventh Text REtrieval Conference (TREC 2002)*. National Institute of Standards and Technology.