Leveraging Wikipedia Characteristics for Search and Candidate Generation in Question Answering

Jennifer Chu-Carroll and **James Fan**
IBM T. J. Watson Research Center
P.O. Box 704, Yorktown Heights, NY 10598, U.S.A.
{jencc,fanj}@us.ibm.com

Abstract

Most existing Question Answering (QA) systems adopt a *type-and-generate* approach to candidate generation that relies on a pre-defined domain ontology. This paper describes a type independent search and candidate generation paradigm for QA that leverages Wikipedia characteristics. This approach is particularly useful for adapting QA systems to domains where reliable answer type identification and type-based answer extraction are not available. We present a three-pronged search approach motivated by relations an answer-justifying title-oriented document may have with the question/answer pair. We further show how Wikipedia metadata such as anchor texts and redirects can be utilized to effectively extract candidate answers from search results without a type ontology. Our experimental results show that our strategies obtained high binary recall in both search and candidate generation on TREC questions, a domain that has mature answer type extraction technology, as well as on Jeopardy! questions, a domain without such technology. Our high-recall search and candidate generation approach has also led to high overall QA performance in Watson, our end-to-end system.

Introduction

Many existing question answering (QA) systems are developed under organized evaluation efforts such as TREC (Voorhees 2002), CLEF (Giampiccolo et al. 2007), and NTCIR (Sasaki et al. 2007). As a result, these systems are developed and evaluated under very similar settings. While the uniform settings facilitate cross-system comparisons, they often result in systems which have similar scope and perform well under similar conditions.

Until recently, QA evaluations have used newswire collections as their reference corpora and focused mainly on factoid questions such as *What is the capital of Japan?* and *How high is Mount Everest?* Many QA systems designed for this task adopt a pipeline that identifies, from a fixed ontology, the expected answer type, retrieves relevant passages from the corpus, then extracts and ranks candidate answers of the right type from the passages. However, in domains where accurate answer type detection and type-based answer extraction are difficult to achieve, the reliance on answer types becomes a hindrance to high QA performance.

To address this issue, we developed an approach to QA independent of a pre-defined ontology, aimed to work effectively across domains (Ferrucci et al. 2010). Our type-independent approach to search and candidate generation leverages the encyclopedic nature of Wikipedia documents and the rich metadata associated with those documents to effectively generate candidate answers with high recall. This paper focuses on how we leverage Wikipedia characteristics and the knowledge inherent in its human-generated metadata to 1) devise a three-pronged search strategy that is more effective than traditional passage search, and 2) develop high recall techniques for producing candidate answers while eliminating the need for hand-crafted type ontologies.

We show that on Jeopardy! questions, where reliable answer type identification and type-based answer extraction are not available, our techniques achieve a search binary recall of 81.4% with 50 document titles and 20 passages and a binary recall of 75.3% for candidates extracted from those search results. On TREC data where approaches based on fixed type ontologies have been effective, we achieve an 80.5% search binary recall and a 73.0% candidate binary recall. These candidates achieve an overall QA accuracy of 59.3% and 49.4% on Jeopardy! and TREC data, respectively, in our end-to-end system, Watson, significantly outperforming almost all TREC QA systems on TREC data. Although discussions in this paper center on Wikipedia, our search strategies apply to other title-oriented corpora and our candidate generation techniques apply to collections of documents with entity-oriented hyperlinks.

Data Analysis

One of the goals of TREC is to achieve open domain QA capability, i.e., systems should be adaptable to new domains with minimal effort. We applied a high performing TREC QA system (Prager et al. 2006) to 2000 Jeopardy! questions[1][2] which span many domains, including arts and entertainment, history, geography, and science. The candidate binary recall (percentage of questions where the correct answer is in the candidate list) on 100 retrieved passages was 39.3%, despite achieving 72.1% on TREC questions.

LAT Analysis

We hypothesize that the discrepancy in candidate recall performance between the two data sets is due to the more

[1] Jeopardy! (http://www.jeopardy.com) is a popular American quiz show that has been on the air for 27 years.

[2] Questions obtained from http://www.j-archive.com.

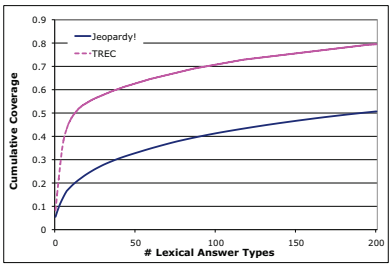

Figure 1: Coverage of top 200 most frequent LATs on Jeopardy! and TREC data

complex language and more diverse answer types in Jeopardy! data. While both data sets contain primarily factoid questions, Jeopardy! questions include much finer grained answer types. Figure 1 shows the cumulative histograms of the 200 most frequent lexical answer types (LATs) for randomly selected Jeopardy! and TREC questions. The LAT is determined by the question focus, such as *president* in *This president is the first 4-star general in the U.S. Army*[3] and indicates the type of the answer sought. Figure 1 shows that while the top 200 TREC LATs cover nearly 80% of all TREC questions, the top 200 Jeopardy! LATs cover only 50% of all those questions. The TREC LAT distribution enables a high-performing QA system to adopt a small ontology with roughly 50-100 semantic answer types. On the other hand, our analysis shows that in 20,000 Jeopardy! questions there were 5,000 distinct LATs, with over 60% singleton occurrences. Although multiple LATs such as *film*, *movie*, and *flick*, can be mapped into a single *movie* semantic type, Jeopardy! data contain many specific LATs like *redhead*, *memoir*, *neo-noir*, and *Ford model*, which, when forced into a typical TREC ontology, would map into a supertype, resulting in substantial loss in information.

Wikipedia Title Coverage Analysis

Although Jeopardy! spans many domains, its questions focus on facts of interest to people. Wikipedia, which contains a wealth of information on such topics, is an excellent reference corpus for these questions. We observed that Jeopardy! answers are often titles of Wikipedia articles. The 5.3% of answers that were not Wikipedia titles often included multiple entities, such as *Red, White & Blue*, or were sentences or phrases, such as *make a scarecrow*. We observed a similar coverage for TREC answers, 12.6% of which were not Wikipedia titles. Excluding numeric answers, however, all but 2% of TREC answers were Wikipedia titles.

Overview of Approach

The observations that Jeopardy! questions contain a large number of distinct LATs and that most answers are Wikipedia titles are key motivations for our multi-pronged approach to search and candidate generation. Our approach, depicted in Figure 2, is answer type independent, a significant departure from most existing QA systems.

[3]Jeopardy! clues are phrased in the form of a statement.

Most existing QA systems adopt a *type-and-generate* approach in which an answer type is determined from the question, and candidate answers of that type are extracted from relevant passages (c.f. (Prager et al. 2000; Moldovan et al. 2000; Yang and Chua 2002)). It presupposes an ontology that covers most answer types in the domain, and a high-performing named entity recognizer for those types. Our large number of LATs and the high ratio of singleton LATs make this approach impractical. Therefore, we eliminated the type ontology pre-requisite and adopted a *generate-and-type* paradigm to QA. In this paradigm, a large number of candidate answers are produced without answer type information and are subsequently evaluated by an ensemble of answer scorers to identify the correct answer. Some answer scorers dynamically compute type matches between a candidate and the LAT using a variety of existing resources (Ferrucci et al. 2010), which contribute to ranking the overall goodness of a candidate answer.

A Three-Pronged Approach to Title-Oriented Search

Motivation

To devise effective search strategies, we examined the relation between a question/answer pair and the title of an answer-justifying Wikipedia document. We identified three relations: when the title is the answer, when the title is in the question, and when it is not the answer or in the question.

In the first scenario, the title of an answer-justifying document is the answer itself. For example, consider *This country singer was imprisoned for robbery and in 1972 was pardoned by Ronald Reagan*. The Wikipedia article for Merle Haggard, the correct answer, mentions him as a country singer, his imprisonment for robbery, and his pardon by Reagan, and is therefore an excellent match for the question. Questions that seek an entity given its well-known attributes often fall into this category. Other examples include seeking a movie given its plot and a company given its products.

In the second scenario, the title of an answer-justifying document appears in the question. For instance, consider *Aleksander Kwasniewski became the president of this country in 1995*. Since articles about countries do not typically contain a list of all its past presidents, the previous approach is likely to fail. However, the first sentence in the Wikipedia page for Aleksander Kwasniewski states: "Aleksander Kwasniewski is a Polish socialist politician who served as the President of Poland from 1995 to 2005." Questions seeking a certain attribute about an entity given in the question often fall into this category. Additional examples include seeking the country of origin given a celebrity and the setting given a play.

In the third scenario, the answer is in a document whose title appears neither as the answer nor in the question. For example, *Shakespeare helped his father obtain this object, with the motto "Non Sanz Droict"*. Passages justifying the correct answer, coat of arms, are neither in the Wikipedia article titled "Coat of Arms", nor in any document whose title is in the question. Rather, they are found in third-party documents, such as "Every Man out of His Humour" and

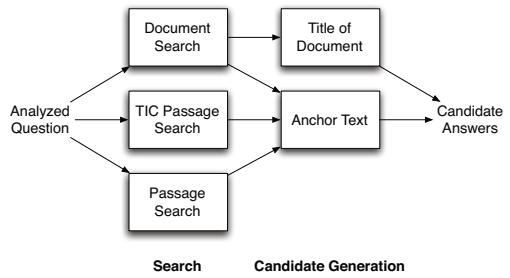

Figure 2: Search and Candidate Generation Components

"Shakespeare's Life". In this case, there is no expectation regarding the relationship between the title of an answer-justifying document and the question/answer themselves.

A Three-Pronged Search Strategy

Motivated by the observation above, we developed search strategies tailored to each of the three relations. We describe our three-pronged search strategy in this section and later show that it outperforms the one-size-fits-all passage search strategy adopted by most existing QA systems.

Our three-pronged search strategy for document and passage retrieval is shown in the left hand column of Figure 2. To achieve high recall, all strategies are attempted for a given question and their results aggregated. **Document Search** aims to retrieve documents which, as a whole, best match information given in the question. It is particularly effective when that information is central to the focus of the document (and is therefore likely to be repeated in the document) or when it contains multiple facts about the answer mentioned in the same document. In these cases, the correct answer is best justified by the totality of the document, which, in encyclopedia documents, corresponds to cases where the answer is the document title.

The **Title in Clue (TIC) Passage Search** and **Passage Search** strategies attempt to identify 1-2 sentence passages that answer the question. The difference between the two strategies lie in the collection of documents from which passages are retrieved. **TIC Passage Search** creates a subcorpus of documents whose titles appear in the question and retrieves passages from that subcorpus. By significantly constraining the search space, TIC Passage Search is effective when the question contains, aside from the identified title(s), primarily common words which are not sufficiently discriminative against a very large collection.

Passage Search is similar to the approach taken by most QA systems in which passages are retrieved from the entire collection. It is comprehensive in coverage and complements the other strategies in addressing cases where answer-justifying passages are present in third-party documents.

Candidate Generation Using Wikipedia Metadata

Motivation

In the candidate generation phase, potential answers are extracted from search results to be scored by downstream components. As discussed earlier, most existing QA systems adopt a *type-and-generate* approach in which a named entity recognizer is employed to identify candidates of the expected type. Our *generate-and-type* approach, on the other hand, requires novel techniques for generating candidates without type information.

Humans have an intuition for what constitutes a plausible answer. Consider the passage, "*Neapolitan pizzas* are made with ingredients like *San Marzano tomatoes*, which grow on the volcanic plains south of *Mount Vesuvius* and *Mozzarella di Bufala Campana*, made with milk from *water buffalo* raised in the marshlands of *Campania* and *Lazio*." Regardless of the question, we expect named entities such as "Neapolitan pizzas", "San Marzano tomatoes", and "Mount Vesuvius" to be plausible candidates. While it is possible to identify proper names with high accuracy, we note that common nouns and verbs can also be candidate answers and that some common nouns and verbs are more plausible candidates than others. For instance, "water buffalo" is a more likely candidate than "grow" and "ingredients".

We previously observed that nearly all answers to our questions are Wikipedia titles. In the above passage, although plausible candidates such as "Mount Vesuvius" and "Lazio" are Wikipedia titles, so are terms like "are" and "raised". In other words, while most answers are Wikipedia titles, only a subset of those titles are plausible answers. We further note that the Wikipedia article titled "Raised" describes it in the context of phonetics, a different word sense than that in the passage. Our analysis of plausible candidates indicate that they often satisfy two criteria. First, they represent salient concepts conveyed in the passage. Second, the candidates have Wikipedia articles about them.

Wikipedia contains metadata, italicized in the above passage, that highly correlate with those two criteria: *anchor texts* and *redirects*. Anchor texts, the clickable text in hyperlinks, represent salient concepts in a document. The targets of hyperlinks are documents highly relevant to the anchor texts. In our sample passage, the anchor texts include all proper names and *water buffalo*, which together capture the desirable candidates without including non-plausible candidates that happen to be Wikipedia titles. Since hyperlinks are used to connect documents, salient concepts best explained by the current document are not linked. To remedy this, we leverage Wikipedia redirect pages, which typically include synonyms and alternative spellings of the title of the target document. For instance, "New York, NY" and "NYC" both redirect to "New York City", and are thus both considered salient concepts in the "New York City" document.

Dual Candidate Generation Strategies

Figure 2 shows the two types of search results, documents and passages, as well as the two candidate generation strategies that can apply to one or both of them.

For document search results, we apply the Title of Document candidate generation strategy where the document title becomes a candidate answer. This strategy is effective for retrieved documents whose title is the answer. However, in some cases the title contains, but is not in itself, the answer. For example, the title of the top ranked document for *The*

Gibson desert can be found on this continent is "Deserts of Australia", which contains the correct answer, "Australia". These substring candidates are generated by the Anchor Text candidate generation strategy, discussed below.

Anchor Text candidate generation implements the extraction of candidate answers using anchor text and redirects metadata from Wikipedia documents. Since salient concepts are often hyperlinked only in its first occurrence in a Wikipedia document, we create an aggregate representation of salient concepts in each document that contains the following information:

1. **Anchor texts** in the document: e.g., "Lazio" and "water buffalo" in our example.
2. **Titles of target documents** of hyperlinks: "Buffalo Mozzarella", , title of document linked to by "Mozzarella di Bufala Campana".
3. **Title of document**: "Pizza".
4. **Titles of redirect pages**: "Neapolitan Pizza", "Pizza Pie", etc., whose target is the current document itself.

Anchor Text candidate generation operates on a retrieved text segment: the title of a document or a passage. In either case, the document from which the text segment was extracted is identified and the salient concepts for that document are retrieved. All salient concepts present in the text segment are extracted as candidate answers.

Experimental Results

Corpus and Data

For our experiments, we used an August 2010 crawl of English Wikipedia, indexed using the Indri search engine.[4] To demonstrate the generality of our approach, we evaluated our system, using the exact same configuration, on two data sets with distinct characteristics. The first is a randomly selected set of 2000 Jeopardy! questions and the second 575 non-numeric factoid questions from the QA track of TRECs 11 and 12.[5] System performance is measured with the **binary recall (BR)** metric, defined as the percentage of questions for which a relevant search result is found in the search phase or for which a correct answer is extracted in the candidate generation phase. This metric is chosen due to our interest in measuring how often these components succeed in bringing the correct answer into the candidate pool for downstream scoring and ranking, and is therefore more appropriate than other metrics (such as MRR) that take into account rudimentary ranking of candidates.

Search Experiments

To evaluate our three-pronged search strategy, an Indri query was constructed from content words and phrases in the question, which constituted the query for Document Search. The query was augmented with the prefix passage[20:6] for Passage Search to retrieve 20-word passages which were then expanded on both ends to sentence boundaries. For TIC Passage Search, the same passage query was issued against a subcorpus of documents, those whose titles appeared in the question. The resulting documents and passages were evaluated individually as well as combined, as shown in Table 1. For computing BR, a document is considered relevant if its title contains the correct answer, and a passage is relevant if it includes the correct answer.

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

[5]Questions with numeric answers are not the focus of the strategies described here. Numeric candidate answers can easily be extracted using, for example, regular expressions.

Search Strategy	Hits	Bytes (approx)	Jeopardy! BR	TREC BR
Passage Srch (BL)	10	2200	64.4%	60.2%
Document Srch	50	1000	66.5%	58.3%
TIC Passage Srch	10	2100	49.8%	47.3%
Combined	70	5300	81.4%	80.5%

Table 1: Search Evaluation Results

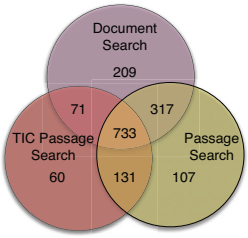

Figure 3: Search Results Overlap Analysis

The Passage Search strategy, similar to that adopted by most existing QA systems, is used as the baseline. Our results on Jeopardy! data show that when retrieving 10 passages, at least one relevant passage is found for 64.4% of the questions. The Document Search strategy, however, achieved a higher BR than the baseline with half the amount of data retrieved. This demonstrates the effectiveness of leveraging the relationship between document titles and content, and underscores the importance of tailoring search strategies to corpus characteristics. Finally, TIC Passage Search retrieved relevant passages for nearly half the questions. While its performance is the lowest of the three strategies, this targeted search approach can succeed when others fail. The last row in the table shows that all strategies combined achieved a BR of 81.4%, a significant improvement over the performance of any single strategy. The TREC results in the table show a similar trend, with the three strategies combined significantly outperforming any single strategy, and obtaining similar performance to Jeopardy! data.

Figure 3 shows an overlap analysis of the three search strategies on the 2000 Jeopardy! questions. All three strategies succeeded in 36.7% of the questions, achieving a reinforcement effect. Furthermore, 18.8% of the questions succeeded with only one strategy, validating the effectiveness of our multi-pronged approach to search. Note that even though TIC Passage Search alone achieved significantly lower performance than the other two strategies, it uniquely succeeded in 60 questions (3.0%), demonstrating the utility of our targeted search strategy. A similar distribution (not shown) was also observed on TREC data.

Candidate Generation Strategy	Jeopardy!		TREC	
	# Candidates	Binary Recall	# Candidates	Binary Recall
Named Entities (baseline)	124.4	54.8%	64.0	52.2%
Wikipedia Titles	273.9	65.5%	133.5	55.7%
Wikipedia Titles (nouns only)	152.2	63.8%	71.3	54.3%
Anchor Text	94.2	64.8%	48.5	52.5%

Table 2: Candidate Generation Evaluation on Passages

Search Result	Candidate Generation Strategy	Jeopardy! Binary Recall		TREC Binary Recall	
		Search	Candidate	Search	Candidate
Document	Title + Anchor Text	66.5%	63.6%	58.3%	55.7%
Passage	Anchor Text	71.0%	64.8%	71.7%	64.4%
All	All	81.4%	75.3%	80.5%	73.0%

Table 3: Candidate Generation Performance on Both Documents and Passages

Candidate Generation Experiments

Our candidate generation experiments focused on the two following aspects: 1) the effectiveness of generating candidates from passages based on document metadata, and 2) the combined overall performance of our three-pronged search approach and dual candidate generation strategies.

Table 2 shows the results of using different candidate generation strategies on the set of 20 passages in Table 1. The baseline, which uses a Named Entity candidate generation strategy, is most directly comparable in BR to most existing QA systems. We ran our rule-based named entity recognizer, which identifies about 200 types primarily motivated by the TREC QA task, on the passages and extracted all entities identified as an instance of any of the recognizer's known types as candidates. Our results on Jeopardy! data show that this baseline has a BR of 54.8%.

Row 2 shows that while BR increased to 65.5% using our initial idea of generating all Wikipedia titles as candidates, more than twice as many candidates were produced. Row 3 shows that by focusing only on Wikipedia titles that are nouns, the number of candidates dropped by 44%, with only minor loss in BR. Finally, Anchor Text candidate generation not only further reduced the number of candidates drastically, but also leveraged the human knowledge inherent in Wikipedia anchor texts and redirects to rival the highest candidate BR for Jeopardy! data.

Table 2 show that Anchor Text candidate generation achieves similar performance as the baseline on TREC data with 25% fewer candidates generated. This validates the effectiveness of our approach on TREC data as well.

Table 3 shows the combined performance of our search and candidate generation components. On Jeopardy! data we obtained roughly 64% candidate BR through document search or passage search alone. When the candidates are pooled, BR reaches 75.3%. For TREC questions, the combined candidate pool achieves a BR of 73.0%, again substantially outperforming either approach alone.

Discussion

Our experimental results show the effectiveness and generality of our approach. Our search strategies, developed to leverage the relationship between title-oriented answer-justifying documents and the question/answer pair, significantly increased search BR on both Jeopardy! and TREC data, compared with the Passage Search strategy alone (Table 1). Our candidate generation strategies represent the most significant departure from existing QA approaches, and focus on utilizing metadata associated with hyperlinked documents to identify salient concepts as plausible answers to questions. In addition to being effective compared against alternative techniques (Table 2), our approach is independent of any fixed, pre-determined type ontology and named entity recognizers, but rather leverages the human knowledge inherent in collaboratively edited Wikipedia documents to extract salient concepts from text.

Table 3 shows that we obtained 75.3% candidate BR on Jeopardy! data and 73.0% on TREC data. We evaluated these candidates in Watson, our end-to-end QA system (Ferrucci et al. 2010), which employs a set of answer scorers to effectively rank these candidates, resulting in an accuracy of 59.3% on Jeopardy! data and 49.4% on TREC data.

Related Work

Most existing QA systems leverage knowledge derived from corpora to enhance performance. To improve search, analysis results from POS taggers, parsers, entity recognizers, and relation detectors have been used to augment the search index (c.f. (Prager et al. 2000; Mihalcea and Moldovan 2001; Katz and Lin 2003; Tiedemann 2005)). Candidate generation techniques also utilized analysis results on passages from entity recognizers and parsers (c.f. (Kupiec 1993; Pasca and Harabagiu 2001; Clarke et al. 2002; Xu et al. 2003)). No system, however, has exploited knowledge intrinsic to the corpus as our system does.

Online encyclopedias such as Grolier and Wikipedia have been used as corpora for several QA systems (Kupiec 1993; Ahn et al. 2004) and in the CLEF evaluation effort (Giampiccolo et al. 2007). However, to our knowledge, these systems treated the new corpus as an extension of the standard newswire corpus and did not exploit its inherent characteristics. In contrast to MacKinnon and Vechtomova (2008), who utilized Wikipedia anchor texts for query expansion in QA, we devised effective QA strategies that lever-

age Wikipedia characteristics and metadata to achieve high search and candidate generation performance.

Wikipedia has been used as a resource for several other NLP and AI tasks, including measuring semantic relatedness (Strube and Ponzetto 2006; Gabrilovich and Markovitch 2007; Milne and Witten 2008; Müller and Gurevych 2008), entity and word sense disambiguation (Mihalcea 2007; Bunescu and Pasca 2006), and knowledge acquisition (Nastase and Strube 2008). As in our work, these studies have found Wikipedia metadata, such as anchor text and category information, to be a useful knowledge source for their tasks.

Conclusions

We described how we leveraged Wikipedia characteristics for search and candidate generation in a QA system, eliminating the fixed type ontology pre-requisite of most existing systems. We developed a three-pronged approach to search based on possible relationships between answer-justifying documents and the question/answer pair, which is generally applicable to title-oriented documents. We devised an effective dual candidate generation strategy that exploits the relationship between encyclopedia article titles and content, and leverages the human knowledge inherent in Wikipedia anchor texts and redirects to identify salient terms as potential candidates. This strategy can generally be applied to documents with entity-oriented hyperlinks. Overall, on a set of 2000 Jeopardy! questions, we achieved an 81.4% BR in search, and a 75.3% BR in candidate generation. These candidates, when ranked by an ensemble of answer scorers, achieved 59.3% in end-to-end QA accuracy. We demonstrated the generality of our approach by showing similar performance characteristics on TREC data with an accuracy that surpasses those of almost all existing QA systems.

Acknowledgments

We would like to thank Chris Welty for providing the LAT analysis results, the Watson team for discussions, and Dave Ferrucci for helpful comments on the paper.

References

Ahn, D.; Jijkoun, V.; Mishne, G.; Muller, K.; de Rijke, M.; and Schlobach, S. 2004. Using Wikipedia at the TREC QA track. In *Proceedings of TREC*.

Bunescu, R., and Pasca, M. 2006. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of EACL*.

Clarke, C.; Cormack, G.; Kemkes, G.; Laszlo, M.; Lynam, T.; E., T.; and Tilker, P. 2002. Statistical selection of exact answers. In *Proceedings of TREC*.

Ferrucci, D.; Brown, E.; Chu-Carroll, J.; Fan, J.; Gondek, D.; Kalyanpur, A.; Adam, L.; Murdock, J. W.; Nyberg, E.; Prager, J.; and Schlaefer, N. 2010. Building Watson: An overview of the DeepQA project. *AI Magazine*.

Gabrilovich, E., and Markovitch, S. 2007. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of IJCAI*.

Giampiccolo, D.; Froner, P.; Peñas, A.; Ayache, C.; Cristea, D.; Jijkoun, V.; Osenova, P.; Rocha, P.; Sacaleanu, B.; and Suteliffe, R. 2007. Overview of the CLEF 2007 multilingual qa track. In *Proceedings of CLEF*.

Katz, B., and Lin, J. 2003. Selectively using relations to improve precision in question answering. In *Proceedings of the EACL Workshop on NLP for QA*.

Kupiec, J. 1993. Murax: A robust linguistic approach for question answering using an on-line encyclopedia. In *Proceedings of SIGIR*.

MacKinnon, I., and Vechtomova, O. 2008. Improving complex interactive question anwering with Wikipedia anchor text. In *Advances in Information Retrieval*.

Mihalcea, R., and Moldovan, D. 2001. Document indexing using named entities. *Studies in Informatics and Control*.

Mihalcea, R. 2007. Using Wikipedia for automatic word sense disambiguation. In *Proceedings of NAACL/HLT*.

Milne, D., and Witten, I. 2008. An effective, low-cost measure of semantic relatedness obtained from Wikipedia links. In *Proceedings of the AAAI Workshop on Wikipedia and AI*.

Moldovan, D.; Harabagiu, S.; Pasca, M.; Mihalcea, R.; Girju, R.; Goodrum, R.; and Rus, V. 2000. The structure and performance of an open-domain question answering system. In *Proceedings of ACL*.

Müller, C., and Gurevych, I. 2008. Using Wikipedia and Wiktionary in domain-specific information retrieval. In *Proceedings of CLEF*.

Nastase, V., and Strube, M. 2008. Decoding Wikipedia categories for knowledge acquisition. In *Proceedings of AAAI*.

Pasca, M., and Harabagiu, S. 2001. High performance question answering. In *Proceedings of SIGIR*.

Prager, J.; Brown, E.; Coden, A.; and Radev, D. 2000. Question-answering by predictive annotation. In *Proceedings of SIGIR*.

Prager, J.; Chu-Carroll, J.; Brown, E.; and Czuba, K. 2006. Question answering using predictive annotation. In *Advances in Open-Domain Question Answering*.

Sasaki, Y.; Lin, C.; Chen, K.; and Chen, H. 2007. Overview of the NTCIR-6 cross-lingual question answering task. In *Proceedings of the 6th NTCIR Workshop*.

Strube, M., and Ponzetto, S. 2006. WikiRelate! computing semantic relatedness using Wikipedia. In *Proceedings of AAAI*.

Tiedemann, J. 2005. Integrating linguistic knowledge in passage retrieval for question answering. In *Proceedings of HLT/EMNLP*.

Voorhees, E. 2002. Overview of the TREC 2002 Question Answering track. In *Proceedings of TREC*.

Xu, J.; Licuanan, A.; May, J.; Miller, S.; and Weischedel, R. 2003. Answer selection and confidence estimation. In *AAAI Spring Symposium on New Directions in QA*.

Yang, H., and Chua, T. 2002. The integration of lexical knowledge and external resources for question answering. In *Proceedings of TREC*.