Passage Retrieval Using Answer Type Profiles in Question Answering

Surya Ganesh Veeravalli and Vasudeva Varma

Language Technologies Research Centre,
IIIT-Hyderabad, India
suryag@research.iiit.ac.in, vv@iiit.ac.in

Abstract. Retrieving answer containing passages is a challenging task in Question Answering. In this paper, we describe a novel passage retrieval methodology using answer type profiles. Our methodology includes two steps: estimation and ranking. In the estimation step, answer type profiles are constructed from question-answer sentence pairs parallel corpus using a statistical alignment model. Each answer type profile consists of triples: the query word, the answering sentence word and the probability of translation. In the ranking step, answer type profiles are incorporated into the Language Modeling framework called Statistical Machine Translation models for Information Retrieval. Using this framework a set of relevant passages are retrieved, given a question. We conducted experiments on FACTOID questions from TREC 2002 to 2006 QA tracks. The experimental results showed significant improvements over different retrieval models including TFIDF, Okapi BM25, Indri and KL-divergence.

Keywords: Question Answering, Passage Retrieval, Statistical Machine Translation Models, Query Expansion

1 Introduction

Question Answering (QA) aims at finding exact answers to natural language questions in a large collection of documents (such as World Wide Web). Compared to a standard document retrieval framework, which just returns relevant documents to a query, a QA system has to respond with an adequate answer to a natural language question. Typically, a QA system has the following four components: 1) Question Analysis, 2) Document Retrieval, 3) Passage Retrieval, and 4) Answer Extraction. The question analysis component analyzes the question to determine its answer type, and to produce a list of keywords. Using these keywords document retrieval searches for a set of potentially relevant documents from the collection. From these documents, passage retrieval selects passages that are likely to contain the answer. Finally, answer extraction searches these passages for the final answer.

Passage Retrieval is considered as one of the key components in a QA system. It reduces the search space for finding the answer from a massive collection of documents to a fixed number of passages (say top 20). Questions which do not have answers in the set of passages considered for answer extraction, cannot be answered correctly by any QA system. So, high performance of passage retrieval is desired to improve the success rate of a QA system. Most often passage retrieval suffers from terminological gap i.e., passages holding the answer to a question have semantic alterations of original terms in the question. Moldovan *et al.* (2003) showed that their system failed to answer 25.7% of questions solely because of terminological gap. This problem is normally addressed by the use of query expansion techniques. These techniques can be broadly classified into two categories; explicit query expansion and implicit query expansion techniques.

In explicit query expansion, new terms are added to the original query to bridge terminological gap between the question and answer containing passages. Different methodologies have been

proposed to expand queries by utilizing top N ranked passages (pseudo-relevance feedback) (Gong *et al.*, 2005) or utilizing external knowledge sources like WordNet, Encyclopedias or Web (Yang *et al.*, 2003). In implicit query expansion, the original query remains unchanged but during the process of retrieval semantic variants of original query terms like their stems (Bilotti *et al.*, 2004) or morphological root forms are considered.

The Statistical Machine Translation (SMT) framework which expands the query implicitly, has been used in several areas of information retrieval (IR). This model was first proposed by Berger and Lafferty (1999) for monolingual document retrieval. In this paper, we describe a passage retrieval methodology leveraging this framework. Our approach includes two phases: one is off-line phase and the other is an on-line phase. The off-line phase constructs answer type profiles (ATPs) from question-answer sentence pairs parallel corpus using a statistical alignment model. Construction of ATPs includes: semantic categorization of questions based on their answer types, and building a distinct translation model for each category (answer type) of questions using a statistical alignment algorithm. These translation models are termed as ATPs. The on-line phase uses ATPs within the SMT framework to retrieve a ranked set of relevant passages given a question.

The rest of this paper is organized as follows: Section 2 describes the related work; Section 3 describes the statistical machine translation model for information retrieval; Section 4 describes our passage retrieval methodology; Section 5 describes the experiments conducted and their results; Section 6 discusses the observations made in experiments and Section 7 concludes the paper.

2 Related Work

Several passage retrieval methodologies have been proposed in the context of QA. Here we briefly overview some of the available methodologies.

Light *et al.* (2001) ranked passages based on the number of terms a passage has in common with the query. Clarke *et al.* (2000) developed a density based passage retrieval algorithm which favors short passages containing many terms with high idf values. In this method the passages are demarcated by query words i.e., each passage starts and ends with a query word. Gonzalez *et al.* (2001) measured non-length normalized cosine similarity between query and the passage. Terms are weighted based on their counts in the passage and in the query, and also their idf values. Cui *et al.* (2005) explored the use of fuzzy dependency relation matching method to enhance passage retrieval by examining dependency relations between query terms and key terms within passages. This approach produced significant improvements when compared to the density based passage retrieval approaches. Similarly Wu *et al.* (2005) extracted surface relation patterns from both the query and the passages to perform relation based matching. The above two techniques are ineffective for short queries which have very less query terms and relation paths. Apart from the above methodologies, there are several other passage retrieval methodologies which used variants of standard retrieval methodologies including vector space models and language modeling.

Murdock and Croft (2005) used the SMT model for sentence retrieval in QA. Their approach used IBM model 1 (Brown *et al.*, 1990) to build a translation model for all the question-sentence pairs in the training corpus. The constructed translation model is used in the language modeling framework to retrieve a ranked set of passages. Their experimental results on TREC data showed that their approach performed better than retrieval based on query likelihood. Our approach for passage retrieval is very similar to the above approach, but we construct more sophisticated multiple translation models which are perceived as answer type profiles i.e., questions from distinct categories (answer types) have distinct translation models. The aim of our passage retrieval approach is that, during retrieval, query words should be expanded inherently with only their contextually related synonyms, where the context is determined by the answer type of the question. For instance, given the question "Where was Paul Krugman born?", our approach aims at searching for contextually related synonyms for the word "born" (such as, "birthplace", "hometown" etc. instead of "birthdate" which is also the synonym of the same word) during retrieval. Hence, during

the process of retrieval, based on the answer type of the question its corresponding answer type profile is used to retrieve relevant passages.

3 Translation Model

The statistical machine translation model or the noisy channel model for IR has found its roots from the statistical language modeling (Ponte, 1998). Language modeling for IR refers to the problem of estimating a probability distribution over the words for each document and calculate the probability that the query is a sample generated from that distribution. So, the documents in the collection are ranked by the probability that a query Q would be generated by the document language model $D : P(Q|D)$. The calculation of this probability differs significantly from model to model. Song and Croft (1999) choose to treat the query Q as a sequence of independent words and hence the documents in the collection are ranked according to the equation below.

$$P(Q|D) = \prod_{q_i \in Q} P(q_i|D)$$

Where the probability $P(q_i|D)$ is estimated by interpolating the term distribution in the document with the term distribution in the collection.

$$P(Q|D) = \prod_{q_i \in Q} \alpha P(q_i|D) + (1-\alpha)P(q_i|C)$$

Where $P(q_i|C)$ is the probability that q_i appears in the collection and α is a weighting parameter which lies between 0 and 1. Berger and Lafferty (1999) has extended this basic language model as a translation model for IR. Within this model the query generation process is viewed as a translation or distillation from a document. To determine the relevance of a document to a query, this model estimates the probability that the query would have been generated as a translation of that document. So for a given query, documents in the collection are ranked according to these probabilities. More specifically, the mapping from a document term w to a query term q_i is achieved by estimating translation models $P(q_i|w)$. Using translation models, the retrieval model becomes

$$P(Q|D) = \prod_{q_i \in Q} \alpha \sum_{w \in D} P(q_i|w)P(w|D) + (1-\alpha)P(q_i|C)$$

Where $P(q_i|w)$ is an entry in the translation model which is typically learned from a parallel corpus consisting of queries and documents relevant to those queries. The learned translation model consists of triples, the query word, the document word and the probability of translation. So, the translation model is a quantified mapping between query words and document words and this mapping could address the problems of synonymy and polysemy in IR. This shows that within the framework of translation models for IR, query is being expanded implicitly.

4 Passage Retrieval

In this section we describe how we performed passage retrieval leveraging the SMT framework. Our methodology includes two steps: estimation and ranking. The estimation step includes: construction of parallel corpus, semantic categorization of questions based on their answer types, and building answer type profiles. In the ranking step, answer type profiles are incorporated into the SMT framework to retrieve a ranked set of passages given a question. The detailed description of individual steps in estimation and ranking are described below.

4.1 Parallel corpus

A parallel corpus consisting of questions and sentences with answers to those questions is required to learn answer type profiles. Kaisser and Lowe (2008) developed a Question Answer Sentence Pair (QASP) corpus to foster research in QA. They identified sentences which contain answers using Amazon's Mechanical Trunk, an "artificial artificial intelligence" web service. The corpus consists of questions from Text Retrieval Conference (TREC) - QA track test sets for the years 2002 to 2006, and sentences consisting of answers from AQUAINT corpus. Table 1 shows the quantitative overview of QASP parallel corpus.

Table 1: Quantitative overview of QASP parallel corpus.

Year	No. factoid questions	No. sentence pairs
2002	429	2,006
2003	354	1,448
2004	204	865
2005	319	1,456
2006	352	1,405

Table 2: The coarse and fine grained answer types.

Coarse	Fine
ABBR	abbreviation, expansion
DESC	definition, description, manner, reason
ENTY	animal, body, color, creation, currency, disease/medical, event, food, instrument, language, letter, other, plant, product, religion, sport, substance, symbol, technique, term, vehicle, word
HUM	description, group, individual, title
LOC	city, country, mountain, other, state
NUM	code, count, date, distance, money, order, other, percent, period, speed, temperature, size, weight

4.2 Question Classification

Construction of an ATP requires the identification of answer type given a question. Earlier approaches for question classification used manually constructed set of rules to map a question to an answer type. These approaches require tremendous amount of tedious work to achieve a reasonable accuracy. So, the focus has been shifted towards machine learning approaches which can automatically construct a high performance question classifier. Zhang and Lee (2003) experimented with five classification algorithms: Nearest Neighbours, Naive Bayes, Decision Tree, Sparse Network of Winnows and Support Vector Machines (SVM). Their experimental results showed that SVM outperformed the other four methods.

For the evaluation of our passage retrieval methodology, we have built a question classifier using SVM. The classifier is trained using a standard data set provided by UIUC (Li and Roth, 2002). It has about 5,500 questions for training and 500 questions for testing which are manually labeled into 6 coarse grained and 50 fine grained answer types in a two level taxonomy (Li and Roth, 2002) as shown in Table 2. The classifier when evaluated for coarse grained classification on 500 test questions, produced an accuracy of 86.8% using bag-of-words as feature.

4.3 Learning Answer type profiles

In our approach, we build multiple translation models, each one for a category (answer type) of questions. We perceive each such translation model as an ATP. Statistical alignment models which maximize the probability of the observed (question, sentence) text pairs using Expectation Maximization algorithm, are used to construct these ATPs. After the maximization process is completed, the word level alignments are set to maximum posterior predictions of the model to produce triples: question word, sentence word, probability. We used GIZA++ (Och and Ney, 2000) an implementation of IBM alignment models (Brown *et al.*, 1993), for building ATPs. Sample profiles for LOCATION and NUMBER types are shown in Table 3 and Table 4 respectively.

Table 3: Translations for word *born* in LOCATION profile

Word	Probability
hometown	0.081337
immigrant	0.0322747
birthplace	0.0244121
competitor	0.0244121
career	0.0242433
birthday	0.0108326

Table 4: Translations for word *born* in NUMBER profile

Word	Probability
born	0.330707
youngest	0.0147641
grandson	0.0147641
nursing	0.0134934
biography	0.00987116
birthdate	0.00492135

4.4 Passage Ranking

This is the ranking or on-line phase of our approach. In this phase, using the ATPs, passages that are relevant to a question are retrieved. To determine the relevance of a passage to a question, the probability that a question would have been generated as a translation of that passage is estimated. Passages are ranked according to these probabilities. The relevance of a passage A returned for question Q with answer type t_j (where $1 \leq j \leq 6$ for coarse grained classification; $1 \leq j \leq 50$ for fine grained classification) is computed using its profile ATP_j as shown in the equation below.

$$\begin{aligned} P(Q|A,t_j) &= \prod_{q_i \in Q} \alpha \sum_{w \in A} P(q_i|w, ATP_j) P(w|A) \\ &\quad + (1-\alpha) P(q_i|C) \end{aligned} \tag{1}$$

Where $P(q_i|w, ATP_j)$ is an entry in the ATP_j, $P(w|A)$ is the probability of word w in the passage A, $P(q_i|C)$ is the probability that q_i appears in the AQUAINT collection and α is the weighting parameter which lies between 0 and 1. In general, passage retrieval depends heavily on the overlap between the query and passage vocabularies. As the aim of our approach is to

overcome the terminological gap problem, we accommodate a special condition which Murdock and Croft (2005) have used for ranking sentences given a question. According to this condition, translations of passage terms to a query term are only considered when the query term is not present in that passage. This is based on the assumption that when a passage already contains query terms then, there would not be any source for terminological gap problem. The mathematical representation of this condition is given in the equation below.

$$\sum_{w \in A} P(q_i|w, ATP_j)P(w|A) = t_i P(q_i|A) + (1 - t_i) \sum_{w \in A} P(q_i|w, ATP_j)P(w|A)$$

Where $t_i = 1$ when $q_i = w$, and 0 otherwise. This condition states that the probability of a query term translating to itself is equal to 1 while ensuring that the translation probabilities sum to one. Passages are finally ranked by accommodating this special condition into equation 1.

5 Experiments

We have used TREC 2002 to 2006 QA data sets to test the effectiveness of our passage retrieval methodology using answer type profiles. These data sets consist of: AQUAINT corpus, factoid questions and answer judgements provided by NIST for these questions. The AQUAINT corpus includes 1,033,461 documents taken from AP newswire, the New York Times newswire and the English portion of the Xinhua News Agency newswire. The documents in this corpus contain paragraph markers which were used as passage level boundaries in our experiments. The answer judgments consist of answer patterns and document ids in which they occur. So, the evaluation is performed under two criteria: strict and lenient. For strict scoring, the answer pattern must occur in the passage, and the passage must be from one of the documents listed as relevant in the answer judgments. For lenient scoring, the answer pattern must occur in the passage.

In our experiments, we used the following metrics for evaluation: Precision at 1, Mean Reciprocal Rank (MRR) at N and Total Document Reciprocal Rank (TDRR) (Bilotti *et al.*, 2004). Precision at 1 measures the proportion of questions for which a correct answer appears in the first retrieved passage. The MRR at N is the mean of the inverse of highest ranked answer bearing passage if that passage appears in the top N. TDRR extends MRR with a notion of recall. It is the sum of all reciprocal ranks of all answer bearing passages per question (averaged over all questions) and attains maximum if all retrieved passages are relevant. In the experiments described below, we considered top 20 passages for evaluation i.e. both MRR at N and TDRR are measured for top 20 passages.

A complete analysis of our passage retrieval methodology was done using the following three experiments.

5.1 Retrieval Models

In this experiment we compared the performance of our passage retrieval methodology against standard retrieval methodologies including vector space models and language models. Two models from vector space models including TFIDF and Okapi BM25, and two models from language modeling including KL-divergence and Indri were selected for comparing the results.

TFIDF: The TFIDF weighting scheme is often used in information retrieval. Many variations of the TFIDF weighting scheme are being used by search engines as a central tool in computing the relevance between a document and a user query. We have used a variant of the TFIDF model based on the Okapi TF formula (Robertson *et al.*, 1996).

Okapi BM25: In information retrieval, Okapi BM25 represents the state-of-the-art retrieval model and is based on the probabilistic retrieval framework developed by Robertson and Walker (1999). It is a ranking function used by search engines to rank matching documents according to their relevance to a given search query.

KL-divergence: The KL-divergence retrieval model (Zhai and Lafferty, 2001) implements the cross entropy of the query model with respect to the document model. It is a standard metric for comparing distributions, which has proved to work well in IR systems.

Indri: The Indri retrieval model is based on a combination of the language modeling and inference network (Turtle and Croft, 1991) retrieval frameworks. Both frameworks, on their own, have been widely studied, applied, and found to be very effective for a wide range of retrieval tasks. Indri combines the benefits of these two frameworks to further enhance retrieval effectiveness of IR systems.

Table 5: Strict evaluation scores for different passage retrieval methodologies. ATP denotes the passage retrieval methodology proposed by us.

Method	Prec@1	MRR	TDRR
TFIDF	0.172	0.255	0.381
Okapi BM25	0.159	0.235	0.348
KL-divergence	0.175	0.255	0.369
Indri	0.177	0.254	0.376
ATP	0.210	0.287	0.430

Table 6: Lenient evaluation scores for different passage retrieval methodologies. ATP denotes the passage retrieval methodology proposed by us.

Method	Prec@1	MRR	TDRR
TFIDF	0.259	0.348	0.705
Okapi BM25	0.227	0.313	0.626
KL-divergence	0.284	0.373	0.750
Indri	0.297	0.381	0.807
ATP	0.311	0.395	0.809

Lemur, a language modeling toolkit provides the implementation of all the above retrieval models. Parameters in all these models were set to default values as provided by the toolkit. Lemur as such does not support passage retrieval. So, we segmented documents in to passages using the paragraph markers. Each such passage is considered as an individual document and indexed separately using the toolkit. A total of five runs were conducted and in each run questions from one of TREC 2002-2006 years were used for testing and the questions from the rest of the years were used to construct ATPs. IBM model 1, which assumes all possible alignments between source sentence and target sentence equally likely, was used to construct ATPs. Similar to earlier works Berger and Lafferty (1999), we have set the α (weighting parameter) value to be 0.95. The average scores of all the five runs were shown in Tables 5 and 6.

5.2 Answer Types

In our methodology, we build a translation model for every category (answer type) of questions. Each such translation model is termed as an ATP and it is used in the SMT framework to retrieve a ranked set of passages given a question. In this experiment we analyzed the performance of our

methodology on different categories (answer types) of questions using a similar setup as that of the previous experiment. Among the four retrieval models considered in the previous experiment, Indri retrieval model performed better. So, we compared its results with the results obtained by using our methodology. Tables 7 and 8 show the average scores for strict and lenient evaluation, and scores for using Indri retrieval model are enclosed in parenthesis.

Table 7: Strict evaluation scores for different categories (answer types) of questions. Scores for Indri retrieval model are enclosed in parenthesis.

Ans. Type	Prec@1	MRR	TDRR
ABBR	0.125 (0.125)	0.156 (0.178)	0.190 (0.220)
DESC	0.200 (0.144)	0.274 (0.225)	0.390 (0.343)
ENTY	0.165 (0.157)	0.235 (0.226)	0.331 (0.323)
HUM	0.198 (0.193)	0.280 (0.272)	0.432 (0.409)
LOC	0.208 (0.215)	0.311 (0.300)	0.494 (0.467)
NUM	0.243 (0.169)	0.309 (0.244)	0.455 (0.348)

Table 8: Lenient evaluation scores for different categories (answer types) of questions. Scores for Indri retrieval model are enclosed in parenthesis.

Ans. Type	Prec@1	MRR	TDRR
ABBR	0.250 (0.125)	0.250 (0.198)	0.468 (0.443)
DESC	0.256 (0.322)	0.343 (0.401)	0.683 (0.836)
ENTY	0.287 (0.278)	0.368 (0.368)	0.747 (0.802)
HUM	0.325 (0.315)	0.421 (0.408)	0.931 (0.927)
LOC	0.369 (0.362)	0.467 (0.444)	1.054 (1.045)
NUM	0.304 (0.262)	0.376 (0.341)	0.688 (0.625)

5.3 Alignment Models

In this experiment we tested the effect of different statistical alignment models on our passage retrieval approach. So, the first experiment is repeated for different statistical alignment models that are used to construct ATPs. We used IBM model 1 and GIZA++ alignment with default parameters to construct these ATPs. IBM model 1 assumes all possible alignments between source sentence and target sentence equally likely and GIZA++ alignment model is a mixture of IBM model1, HMM alignment model, IBM model3 and IBM model4. The average strict and lenient scores when a particular alignment model is used, are shown in Table 9.

Table 9: Strict and lenient evaluation scores for different statistical alignment models

Strict Evaluation

Model	Prec@1	MRR	TDRR
IBM Model1	0.210	0.287	0.430
GIZA++	0.210	0.286	0.431

Lenient Evaluation

Model	Prec@1	MRR	TDRR
IBM Model1	0.311	0.395	0.809
GIZA++	0.304	0.388	0.802

6 Discussion

Typical retrieval methodologies in IR like vector space models and language modeling directly match the exact query terms on to the documents. Whereas, a statistical machine translation model for IR leverages a precomputed mapping between query terms and document/passage/sentence terms to quantify the relevance of a document given a query. Such a mapping solves the problem of synonymy in IR. So, the application of the SMT model for IR resulted in significant improvements compared to standard retrieval models like TFIDF.

Along with synonymy, our methodology also addresses the problem of polysemy i.e., a term has different meanings in different contexts. We solve this problem by constructing distinct translation models for distinct categories (answer types) of questions. For example given the questions *Q1: When was Paul Krugman born?* and *Q2: Where was Paul Krugman born?*, our methodology uses NUMBER profile for Q1 and LOCATION profile for Q2. Looking at the NUMBER and LOCATION profiles for the word *born* in Tables 3 and 4 respectively, we can observe that the word *born* is mapped to location related terms with high probabilities in LOCATION profile and date related terms in NUMBER profile. So, this infers that our methodology of using multiple translation models addresses the problem of polysemy.

Results from the first experiment showed that our approach outperformed other standard retrieval models including vector space models and language modeling, especially for strict evaluation. And, among the retrieval models considered in the first experiment, Indri performed better, which is a stat-of-the-art retrieval methodology used for both document and passage retrieval. These improvements can be attributed to the ability of our methodology to overcome the problems of synonymy, and polysemy to some extent.

Our analysis on the performance of our passage retrieval methodology on different coarse grained categories of questions showed better improvements for NUM type questions. We believe this is because, a large fraction of questions from TREC 2002-2006 data sets are NUM type questions, which facilitated the construction of highly accurate profile. From the third experiment we found that simple alignment models like IBM model 1 performed better than GIZA++ alignment with default parameters. We believe this is because, IBM model 1 is more suited for IR because the subtler aspects of language used for machine translation can be ignored for IR.

7 Conclusion

Passage retrieval is a key component in a QA system. Unlike typical passage retrieval methodologies which match the exact query terms on to the passages, our methodology leverages the SMT framework. In this framework, a precomputed mapping between query terms and passage terms, is used to rank passages given a question. Our methodology does not rely on any external knowledge sources like WordNet, Encyclopedias or Web to enhance the passage retrieval performance. Instead, it uses previously answered questions and their answering sentences data to rank passages given a question. So, this can be considered as an alternative passage retrieval methodology.

We conducted experiments on TREC 2002-2006 QA data sets. These experiments showed that our methodology outperformed standard retrieval methodologies including TFIDF, Okapi BM25, KL-divergence and Indri. We found that simple statistical alignment models like IBM model 1 are more suited for passage retrieval in QA. We also showed that our methodology addresses the problems of synonymy and polysemy in IR. In the future, we would like to investigate the impact of question classification accuracy and different statistical alignment models on the performance of our retrieval methodology.

References

Berger, A. and J. Lafferty. 1999. Information retrieval as statistical translation. In *SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 222–229, New York, NY, USA. ACM.

Bilotti, M., B. Katz, and J. Lin. 2004. What works better for question answering: Stemming or morphological query expansion? In *ACM SIGIR'04 Workshop Information Retrieval for QA*.

Brown, P. F., J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer, and P. S. Roossin. 1990. A statistical approach to machine translation. *Comput. Linguist.*, 16(2):79–85.

Brown, P. F., V. J. Della Pietra, S. A. Della Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19:263–311.

Clarke, C. L. A., G. V. Cormack, and E. A. Tudhope. 2000. Relevance ranking for one to three term queries. *Inf. Process. Manage.*, 36(2):291–311.

Cui, H., R. Sun, K. Li, M.-Y. Kan, and T.-S. Chua. 2005. Question answering passage retrieval using dependency relations. In *SIGIR '05: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 400–407, New York, NY, USA. ACM.

Gong, Z., C. W. Cheang, and L. H. U. 2005. Web query expansion by WordNet. In *Proceedings of the 16th International Conference on Database and Expert Systems Applications (DEXA 2005)*, Volume 3588 of *Lecture Notes in Computer Science*, pages 166–175. Springer.

Gonzalez, J. L. V., A. F. Rodriguez, and F. Llopis. 2001. University of Alicante at TREC-10. In *Proceedings of TREC-10*.

Kaisser, M. and J. Lowe. 2008. Creating a research collection of question answer sentence pairs with amazon's mechanical turk. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2008/.

Li, X. and D. Roth. 2002. Learning question classifiers. In *Proceedings of the 19th International Conference on Computational Linguistics*, pages 556–562.

Light, M., G. S. Mann, E. Riloff, and E. Breck. 2001. Analyses for elucidating current question answering technology. *Nat. Lang. Eng.*, 7(4):325–342.

Moldovan, D., M. Paşca, S. Harabagiu, and M. Surdeanu. 2003. Performance issues and error analysis in an open-domain question answering system. In *ACM Trans. Inf. Syst*.

Murdock, V. and W. B. Croft. 2005. A translation model for sentence retrieval. In *HLT '05: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 684–691, Morristown, NJ, USA. Association for Computational Linguistics.

Och, F. J. and H. Ney. 2000. Improved statistical alignment models. In *ACL '00: Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 440–447, Morristown, NJ, USA. Association for Computational Linguistics.

Ponte, J. M. 1998. A language modeling approach to information retrieval. Master's thesis, University of Massachusetts, Amherst, MA, USA.

Robertson, S., S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford. 1996. Okapi at TREC-3. In *Proceedings of TREC-3*, pages 109–126.

Robertson, S. E. and S. Walker. 1999. Okapi/keenbow at TREC-8. In *TREC*.

Song, F. and W. B. Croft. 1999. A general language model for information retrieval. In *CIKM '99: Proceedings of the Eighth International Conference on Information and Knowledge Management*, pages 316–321, New York, NY, USA. ACM.

Turtle, H. and W. B. Croft. 1991. Evaluation of an inference network-based retrieval model. *ACM Trans. Inf. Syst.*, 9(3):187–222.

Wu, M., M. Duan, S. Shaikh, S. Small, and T. Strzalkowski. 2005. University of Albany's ILQUA in TREC 2005. In *Proceedings of TREC-14*, pages 77–83.

Yang, H., T.-S. Chua, S. Wang, and C.-K. Koh. 2003. Structured use of external knowledge for event-based open domain question answering. In *SIGIR '03: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 33–40, New York, NY, USA. ACM.

Zhai, C. and J. Lafferty. 2001. Model-based feedback in the language modeling approach to information retrieval. In *CIKM '01: Proceedings of the Tenth International Conference on Information and Knowledge Management*, pages 403–410, New York, NY, USA. ACM.

Zhang, D. and W. S. Lee. 2003. Question classification using support vector machines. In *SIGIR '03: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, pages 26–32, New York, NY, USA. ACM.