Exploiting the use of Prior Probabilities for Passage Retrieval in Question Answering

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

Abstract

Document Retrieval assumes that a document is independent of its relevance, and non-relevance. Previous works showed that the same assumption is being considered for passage retrieval in the context of Question Answering. In this paper, we relax this assumption and describe a method for estimating the prior of a passage being relevant, and non-relevant to a question. These prior probabilities are used in the process of ranking passages. We also describe a trivial method for identifying relevant and non-relevant text to a question using the Web and AQUAINT corpus as information sources. An empirical evaluation on TREC 2006 Question Answering test set showed that in the context of Question Answering prior probabilities are necessary in ranking the passages.

1 Introduction

Passage Retrieval is an intermediate step between document retrieval and answer extraction in a typical Question Answering (QA) system. It reduces the search space for finding an answer from a massive collection of documents to a fixed number of passages (say top 100). Unless the answer is present in one of the retrieved passages, QA systems will not find the answer to a given question. So, passage retrieval is considered as one of the most important components of a QA system.

The Probability Ranking Principle [13] states that a retrieval system should rank the documents in decreasing order of their probability of relevance to the query. According to the Language Modeling [11] decomposition [8] of this ranking principle, the documents should be ranked using the following equation:

$$\log rank(D) = \log p(Q|D, R) + \log \frac{p(D|R)}{p(D|N)} \qquad (1)$$

Here the first term $p(Q|D, R)$ measures the likelihood of the query given a document that is relevant and Language Modeling is being used to estimate this value. The second term measures the prior probabilities of document being relevant, and non relevant. But, document retrieval assumes that a document is independent of its relevance, and non-relevance. So, documents are just ranked based on Language Modeling i.e., the probability of a query being generated by a document. Previous works [9] [10] showed that the same approach is being used even for passage retrieval in the context of QA.

Previously Jagadeesh et al. [5] used prior probabilities in Query-Based Multi-Document Summarization task. They defined an entropy based measure called Information Measure to capture the prior of a sentence. This information measure was computed using external information sources like the Web and Wikipedia. Their experimental results showed that prior probabilities are necessary for ranking sentences in the summarization task. We use a similar approach to exploit the use of prior probabilities for passage retrieval in QA.

In this paper we describe a mutual information measure called KullbackLeibler divergence (KL divergence) [3] to compute the prior of a passage. We also describe a trivial method for identifying relevant and non-relevant text to a question using the Web and AQUAINT corpus (used in TREC[1] QA evaluations) as information sources. The rest of this paper is organized as follows: Section 2 describes the estimation of prior probabilities of passages; Section 3 describes the identification of relevant and non-relevant text to a question; Section 4 describes the experiments conducted and their results and Section 5 concludes the paper.

2 Estimation of prior probability

In this section we assume that relevant (R) and non-relevant (N) text is identified for a given question. In Information Retrieval, KullbackLeibler divergence is often used to measure the distance between two language models [2] [14]. We use this mutual information measure to estimate prior probabilities of passages. Let U_A denotes the unigram language model of passage A and U_R, U_N denote the unigram language models of relevant and non-relevant text respectively. KL divergence between U_A, U_R and U_A, U_N are computed as follows:

$$D(U_A||U_R) = \sum_{v \in V} U_A(v) \log \frac{U_A(v)}{U_R(v)}$$

$$D(U_A||U_N) = \sum_{v \in V} U_A(v) \log \frac{U_A(v)}{U_N(v)}$$

[1] Text REtrieval Conference, http://trec.nist.gov

Where v is a term in the vocabulary V and $U_A(v)$, $U_R(v)$, $U_N(v)$ are the unigram probabilities of v in the passage, relevant and non-relevant text respectively. With the increase in the divergence between passage and relevant text, the probability of passage being relevant decreases. So, the prior probabilities are estimated as follows:

$$p(A|R) = \frac{1}{1 + D(U_A||U_R)}$$

$$p(A|N) = \frac{1}{1 + D(U_A||U_N)}$$

As KL divergence is always non-negative, both $p(A|R)$ and $p(A|N)$ always lie in the range $[0, 1]$. This satisfies the basic law of probability i.e., the probability of an event should always lie in the range $[0, 1]$. $p(A|R) = 1$ when $U_A = U_R$, as the divergence of two equivalent distributions is zero. Similarly, $p(A|N) = 1$ when $U_A = U_N$. Substituting the above estimates for prior probabilities in equation 1 gives the final ranking ranking function for passage retrieval.

$$\log rank(A) = \log p(Q|A, R) - \log \frac{1 + D(U_A||U_R)}{1 + D(U_A||U_N)}$$

3 Identifying relevant and non-relevant text

In the previous section we have assumed that the relevant and non-relevant text for a given question is known. Here we will discuss a method to extract the required information based on different query formulation strategies.

3.1 Relevant text

Breck et al. [1] noticed a correlation between the number of times an answer appeared in the TREC corpus and the average performance of TREC systems on that particular question. This shows that, the more times an answer appears in the text collection, the easier it is to find it. As a text collection, the Web is larger in size than any research corpus by several orders of magnitude. An important implication of this size is the amount of data redundancy inherent in the Web i.e., each item of information has been stated in a variety of ways in different documents in the Web.

Data redundancy in the Web indicates that the answer for a given natural language question exists in many different forms in different documents. So, our methodology for extracting relevant text relies on Web search engines. Currently, the Yahoo search engine is used to retrieve this text from the Web. Assuming that an answer is likely to be found within the vicinity of set of keywords in the question, a query composed of keywords in it is given to the search engine. For example, given the question *"Which position did Warren Moon play in professional football?"*, the following query *"position warren moon play professional football"* is given to the search engine. The top N snippets/summaries provided by the search engine are extracted to form relevant text.

Most of the snippets provided by the search engine consist of broken sentences. These broken sentences may miss a part of answer pattern or entire answer pattern which is originally present in them. In either case, an automatic evaluation using a set of questions and their corresponding answer patterns will fail to show the actual quality of snippets. So, we manually examined the snippets for a set of 50 randomly selected questions from TREC 2006 test set [4]. We observed that on an average about 6 snippets out of top 10 snippets provided by the search engine are relevant to the question. As the quality of snippets is considerably high, we use them as relevant text to a given question.

3.2 Non-relevant text

The methodology for extracting non-relevant text is independent of the size of a text collection unlike the methodology for relevant text. Here the structure of a question is used to extract the required information. An input question is parsed to get POS tags corresponding to all the terms in it. We have used the stanford parser [6] [7] to get POS tag sequence corresponding to a question. Based on POS tags, all keywords in a question are splitted into two sets: Topic and Keyword.

> **Topic:** Typically, questions ask for a specific information within a broad topic. For example, the question *"Which position did Warren Moon play in professional football?"*, asks for a specific information regarding *"Warren Moon"*. A topic can be a person, location, organization, event or any other entity, which are proper nouns. So, a topic set consists of all the proper nouns within a question. And, in questions where there are no proper nouns like *"Which country is the leading producer of rice?"*, nouns *"rice"* and *"country"* are considered as individual topics and these terms form topic set.
>
> **Keywords:** This set contains all the keywords in a question which are not members of topic set. So, for the question *"Which position did Warren Moon play in professional football?"*, the constituents of this set are "*position*", "*play*", "*professional*" and "*football*".

Using the above two sets, two distinct queries are formulated which represent their non-relevance to a question.

> **QUERY I:** It is formulated using topic set terms alone, which is based on the idea that text which covers general information regarding a topic in the question can be considered as non-relevant to it. So, for the above example question "*warren moon*" is expected to retrieve non-relevant text.
>
> **QUERY II:** It is formulated using terms from both topic and keyword sets. The idea behind this query formulation is that text which covers information about a topic in the question but does not contain any of the keywords in it, can be considered as non-relevant to it. So, for the above

example question, "*warren moon -position -play -professional -football*" is expected to retrieve non-relevant text. The negative operator (-) in the above query restricts the Information Retrieval system to retrieve only information without terms in the query that succeed '-' operator.

As the methodology is independent of the size of a corpus, two text collections which include Web and AQUAINT corpus, are used to extract the required information. An empirical evaluation using TREC 2006 QA test set was performed to test the quality of text extracted by using the two queries described previously. Redundancy, a passage retrieval performance evaluation metric, is used to measure the average number of answer bearing passages found within the top N passages retrieved for each query formulation. So, here the quality of text is inversely proportional to redundancy i.e., lower the redundancy value better is the quality of text extracted. All the FACTOID questions from the test set were used to measure redundancy. Table 1 shows the average redundancy scores for the top N passages retrieved from AQUAINT corpus in the test set. QUERY I and QUERY II are the query formulations from a question as described previously and QUERY is a keyword query formulated for retrieving relevant snippets from the Web. These results show that QUERY II produces better quality of non-relevant text than QUERY I. And, compared to QUERY both QUERY I and QUERY II have significantly lower redundancy scores. A similar evaluation could not be performed on snippets retrieved from Web because of broken sentences as described in the previous section.

Query	Top 1	Top 10	Top 20	Top 100
QUERY	0.222	0.844	1.202	2.227
QUERY I	0.020	0.116	0.236	0.597
QUERY II	0.006	0.057	0.122	0.270

Table 1: *Redundancy scores for the passages retrieved from AQUAINT corpus using different queries*

As the extracted relevant and non-relevant text is not truly relevant and non-relevant to a question, a linear interpolation of Language Modeling score and prior probabilities are used to rank passages as shown in the equation below.

$$\log rank(A) = (1-\alpha)\log p(Q|A,R) \\ -\alpha \log \frac{1+D(U_A||U_R)}{1+D(U_A||U_N)}$$

Where α is a weighting parameter which lies between 0 and 1.

4 Experiments

In the context of QA, coverage and redundancy [12] are the two principal measures used to measure the performance of passage retrieval. The coverage gives the proportion of questions for which a correct answer can be found within the top N passages retrieved for each question. The redundancy gives the average number of answer bearing passages found within the top N passages retrieved for each question. In our experiments we have set N as 20 i.e., the top 20 passages are used for evaluation.

The data used to test the effectiveness of prior probabilities of passages includes: AQUAINT corpus, factoid questions from TREC 2006 QA task, and answer judgments provided by NIST for these questions. The AQUAINT corpus consists of 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 are used as passage level boundaries for our experiments. The answer judgments consist of answer patterns and document ids in which they occur. This allows the evaluation to be 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.

We used two open source retrieval engines, Lucene and Indri, to test the effect of prior probabilities on passage retrieval. Lucene supports Boolean query language and ranked retrieval using BM25. Indri is a state-of-the-art retrieval engine that combines the merits of language model and inference network. We incorporated our approach for passage retrieval as a re-ranking step into these retrieval engines. After Lucene or Indri retrieves a ranked set of passages for a given question, top 200 passages are re-ranked, of which top 20 passages are considered for evaluation. The scores for top 20 passages returned by respective engines act as baseline to compare the re-ranked results using our approach.

We performed two experiments in which QUERY and QUERY II were used to extract relevant and non-relevant text respectively. In the first experiment, we compared the re-ranked and baseline results from the two retrieval engines, and they are shown in tables 2 and 3. Only Web was used to extract relevant text but for extracting non-relevant text both AQUAINT and Web were used. So, to analyze the effect of two text collections on computing the prior of a passage, we showed results for both of them. The results listed under AQUAINT and Web show considerable improvements over the baseline and in between the two, scores are marginally higher when Web was used.

Criteria	Metric	Lucene	AQUAINT	Web
Strict	Coverage	0.597	0.639	0.662
	Redundancy	1.202	1.313	1.341
Lenient	Coverage	0.719	0.770	0.781
	Redundancy	3.514	3.790	3.957

Table 2: *Lucene evaluation results under strict and lenient criteria*

In the second experiment we tested our methodology for different values of weighting parameter (α) between 0.0 and 1.0 in the ranking function. Figure 1 shows the performance of passage retrieval for differ-

Criteria	Metric	Indri	AQUAINT	Web
Strict	Coverage	0.548	0.554	0.582
	Redundancy	1.043	1.074	1.114
Lenient	Coverage	0.685	0.707	0.719
	Redundancy	3.349	3.514	3.730

Table 3: *Indri evaluation results under strict and lenient criteria*

ent α values under strict and lenient criteria. In all the cases, the performance of passage retrieval improves over the baseline ($\alpha = 0.0$) for α values between 0.0 and 0.8, and from then it is below the baseline. And, the performance reaches maximum for α values between 0.3 and 0.5 which shows that performance is biased towards query likelihood scores.

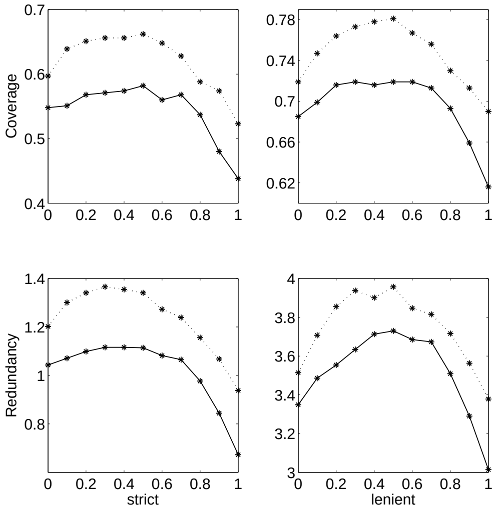

Fig. 1: *Performance of passage retrieval for different α values from 0.0 to 1.0 under strict and lenient criteria. In all the cases '(—*—)' and '($\cdots$ * $\cdots$)' denotes re-ranked scores from Indri and Lucene.*

5 Conclusion

Question Answering aims at finding exact answers to natural language questions from a large collection of documents. Within a QA system, passage retrieval reduces the search space for finding an answer from such large collection of documents to a fixed number of passages. In this paper, we have explored the use of prior probabilities of a passage being relevant, and non-relevant to a question in the process of ranking passages. We described a method for estimating these prior probabilities using KullbackLeibler divergence, and a method for extracting relevant and non-relevant text to a question.

Our experiments on factoid questions from TREC 2006 test set showed that in the context of QA, use of prior probabilities improves the performance of passage retrieval. The experimental results also showed that performance is biased towards query likelihood scores. This could be because the information used for computing prior of a passage is not strictly relevant or non-relevant. In the future, we aim to further enhance the performance of our passage retrieval methodology by exploring different text classification algorithms to derive better prior probability estimates, and different techniques to extract relevant and non-relevant information to a question.

References

[1] E. Breck, M. Light, G. S. Mann, E. Riloff, B. Brown, P. Anand, M. Rooth, and M. Thelen. Looking under the hood : Tools for diagnosing your question answering engine. *CoRR*, cs.CL/0107006, 2001.

[2] C. Carpineto, R. de Mori, G. Romano, and B. Bigi. An information-theoretic approach to automatic query expansion. *ACM Trans. Inf. Syst.*, 19(1):1–27, 2001.

[3] T. M. Cover and J. A. Thomas. *Elements of information theory*. Wiley-Interscience, New York, NY, USA, 1991.

[4] H. T. Dang, J. J. Lin, and D. Kelly. Overview of the trec 2006 question answering track 99. In *TREC*, 2006.

[5] J. Jagarlamudi, P. Pingali, and V. Varma. Capturing sentence prior for query-based multi-document summarization. In D. Evans, S. Furui, and C. Soulupuy, editors, *RIAO*. CID, 2007.

[6] D. Klein and C. D. Manning. Accurate unlexicalized parsing. In *ACL '03: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 423–430, Morristown, NJ, USA, 2003. Association for Computational Linguistics.

[7] D. Klein and C. D. Manning. Fast exact inference with a factored model for natural language parsing. In *In Advances in Neural Information Processing Systems 15 (NIPS)*, pages 3–10. MIT Press, 2003.

[8] J. Lafferty and C. Zhai. *Probabilistic Relevance Models Based on Document and Query Generation*, volume 13. Kluwer International Series on Information Retrieval, 2003.

[9] X. Liu and W. B. Croft. Passage retrieval based on language models. In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 375–382, New York, NY, USA, 2002. ACM.

[10] V. Murdock and W. B. Croft. 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, 2005. Association for Computational Linguistics.

[11] J. M. Ponte. A language modeling approach to information retrieval. Master's thesis, Amherst, MA, USA, 1998.

[12] I. Roberts and R. Gaizauskas. Evaluating passage retrieval approaches for question answering. In *In Proceedings of 26th European Conference on Information Retrieval*, pages 72–84, 2003.

[13] S. E. Robertson. The probability ranking principle in ir. pages 281–286, 1997.

[14] C. Zhai and J. Lafferty. 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, 2001. ACM.