T2R2 東京科学大学 リサーチリポジトリ
Science Tokyo Research Repository

論文 / 著書情報
Article / Book Information

Author	Matthias Heie, Edward Whittaker, Josef Novak, Sadaoki Furui
Journal/Book name	IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU2007), Vol. , No. , pp. 219-224
発行日 / Issue date	2007, 12
権利情報 / Copyright	(c)2007 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Powered by T2R2 (Science Tokyo Research Repository)

A LANGUAGE MODELING APPROACH TO QUESTION ANSWERING ON SPEECH TRANSCRIPTS

Matthias H. Heie, Edward W. D. Whittaker, Josef R. Novak and Sadaoki Furui

Tokyo Institute of Technology, Department of Computer Science
2-12-1 Ookayama, Meguro-ku, Tokyo, 152-8552 Japan,
{heie,edw,novakj,furui}@furui.cs.titech.ac.jp

ABSTRACT

This paper presents a language modeling approach to sentence retrieval for Question Answering (QA) that we used in Question Answering on speech transcripts (QAst), a pilot task at the Cross Language Evaluation Forum (CLEF) evaluations 2007. A language model (LM) is generated for each sentence and these models are combined with document LMs to take advantage of contextual information. A query expansion technique using class models is proposed and included in our framework. Finally, our method's impact on exact answer extraction is evaluated. We show that combining sentence LMs with document LMs significantly improves sentence retrieval performance, and that this sentence retrieval approach leads to better answer extraction performance.

Index Terms— Question Answering, Sentence Retrieval, Language Modeling

1. INTRODUCTION

Question Answering (QA), defined as the task of answering questions posed in natural language, has attracted considerable research interest since the introduction of a QA track in the Text REtrieval Conference (TREC) evaluations in 1999 [1]. Until now research in QA has focused on extracting answers from written text. However, the most natural means of human communication is speech, and a large amount of speech data is already available, for example in radio and TV broadcast archives, and much more is obtainable by recording lectures, seminars, meetings, etc. All these sources provide information that could be mined for QA purposes.

In 2007, the Cross Language Evaluation Forum (CLEF) introduced as a pilot task the QAst track [2], where answers to written questions had to be extracted from spoken data. The task was restricted to factoid questions, i.e. questions that are answered with a few words and typically use question words such as *"what"*, *"when"*, *"where"*, *"who"*, etc. Although there has been some previous research in QA on speech transcripts [3, 4], most effort on bringing speech into a QA scenario has been focused on providing users of QA systems with a speech interface [5].

The transition from using text corpora to using speech corpora is not trivial. In addition to the traditional challenges of QA, the difficulties of spoken language understanding, such as word errors, ungrammatical constructions, repetitions, hesitations, fillers, etc., must also be handled, hence a robust approach is required. Moreover, the QAst track posed additional challenges to our approach in previous QA evaluations, where we have used much larger corpora for finding answers. For example, in previous TREC evaluations around 1 million documents were available, allowing us to take advantage of redundancy. In contrast, the QAst evaluation corpus contained only 15 lecture transcripts, with little or no redundancy for most questions.

For the QAst evaluation we used an entirely data-driven QA framework, designed with language independence in mind, thus no explicit linguistic knowledge was utilized. The framework has a two-stage QA architecture, common in contemporary QA systems. In the first stage the information retrieval (IR) module retrieves passages likely to contain the correct answer from a collection of lecture transcripts. In the second stage the answer extraction (AE) module extracts the exact answer from the retrieved passages. We used a language modeling approach to IR, and defined a passage to be a sentence. There were two major reasons why we chose to work with sentences. Firstly, we envision an implementation of a speech-driven QA system where the user is played back a snippet of speech that should contain the answer. In this scenario the user would most likely prefer to hear not only the few words that constitute the answer, but also the immediate context in which the answer appears, similar to how modern search engines provide users with the context the query terms appear in. This way the user can evaluate the supporting evidence of the answer. Thus extracting the answer from the sentence would not be necessary, and the answer words do not even need to have been correctly recognized. Secondly, while many QA frameworks take advantage of redundancy in the corpus [6], there are situations where little or no redundancy is available. As stated above, this was the case in the QAst task, where in most cases the answer to a question appeared only once in the corpus. Under such circumstances

(and given that the precise answer with no context is required) we believe it is preferable to search for the answer in one or a few sentences which have a high probability of containing the answer, rather than supplying the AE module with a larger amount of more noisy data as we have done in previous QA evaluations.

The rest of the paper is organized as follows. We start by explaining the methods we employed in our QA framework (Section 2), then the experimental framework (Section 3), results (Section 4) and discussion (Section 5), and finally conclusions (Section 6).

2. METHODS

2.1. Information retrieval using language modeling

The general approach to IR for QA is to treat each question word as a query term, but disregard question-type words such as *"what"*, *"when"*, *"who"*, etc., and possibly also a set of stop words, typically frequent less-informative words such as *"is"*, *"have"*, etc. Of the many ways to model the IR task, language modeling has gained much interest over the last decade since the approach was proposed [7]. Under this approach a LM is estimated for each document. The documents are then ranked according to the conditional probability $P(Q|D)$, the probability of generating the query Q given the document D.

A language model based approach to IR for QA is presented in [8]. Here a special case of passage retrieval called sentence retrieval is used, where each passage contains only one sentence. Due to lack of data to train the sentence specific LM, it is assumed that all words are independent, hence unigrams are used:

$$P(Q|S) = \prod_{i=1}^{|Q|} P(q_i|S), \quad (1)$$

where q_i is the ith query term in the query $Q = (q_1 ... q_{|Q|})$ composed of $|Q|$ query terms. Throughout this paper we calculate the probability of a query term q given a sentence S in three different ways: $P_1(q|S)$, $P_2(q|S)$ and $P_3(q|S)$, as explained below.

Smoothing methods are normally employed with LMs to avoid the problem of zero probabilities when one of the query terms does not occur in the document. This is typically achieved by redistributing probability mass from the document model to a background collection model $P(Q|B)$. We use absolute discounting, where the probability of a query term q given a sentence S is calculated as:

$$P_1(q|S) = \frac{\max\{tf(q,S) - \delta, 0\}}{l(S)} + \frac{\delta \cdot h(S,\delta)}{l(S)} \cdot P(q|B), \quad (2)$$

where $tf(q,S)$ is the term frequency of q in S, $l(S)$ is the length (number of words) of S, δ is the discount parameter, $h(S,\delta)$ is the count of how many unique words in S have a term frequency higher than δ, and $P(q|B)$ is the unigram probability of the query term q according to the background collection model. Note that if $\delta < 1$ then $h(S,\delta)$ is equal to the number of unique words in S.

A problem with the model presented in [8] is that words relevant to the sentence might not occur in the sentence itself, but in the surrounding text. For example, for the question *"In which city was the 93 Eurospeech conference held?"*, the sentence *"In 93 the conference was held in Berlin."* in an article about Eurospeech should ideally be assigned a high probability, despite the sentence missing the word *"Eurospeech"*. To account for this, we train document LMs, $P_1(q|D)$, in the same manner as for $P_1(q|S)$ in Eq. (2), and perform a linear interpolation between $P_1(q|S)$ and $P_1(q|D)$:

$$P_2(q|S) = (1-\alpha) \cdot P_1(q|S) + \alpha \cdot P_1(q|D), \quad (3)$$

where $0 \leq \alpha \leq 1$ is an interpolation parameter.

2.2. Query expansion

Query expansion, which has been shown to improve IR performance also for QA tasks [9], involves adding new terms to the initial query that are semantically close to the original terms. Techniques for query expansion fall into two general categories: global methods and local methods. Global methods expand the query based on collection data, while local methods perform an initial retrieval and expand the query based on the top ranked documents. We experiment with a global method in which words are grouped into a set $C = \{c_1 ... c_{|C|}\}$ of $|C|$ overlapping classes beforehand, and calculate the unigram class model probability of a query term q given a sentence S as follows:

$$P_C(q|S) = \sum_{j=1}^{|C|} P(q|c_j) \cdot P(c_j|S), \quad (4)$$

where $P(q|c_j) = 1/|c_j|$ if $q \in c_j$, else $P(q|c_j) = 0$, where $|c_j|$ is the number of words in c_j. $P(c_j|S)$ can be re-written as a sum over the $|V|$ words in the vocabulary $V = \{w_1 ... w_{|V|}\}$:

$$P(c_j|S) = \sum_{k=1}^{|V|} P(c_j|w_k) \cdot P(w_k|S), \quad (5)$$

where $P(c_j|w_k) = 1/N(w_k, C)$ if $w_k \in c_j$, else $P(c_j|w_k) = 0$. $N(w_k, C)$ is the number of classes in C where w_k occurs. $P(w_k|S)$ is the unigram probability of the word w_k given the sentence S.

The word LM in Eq. (2) and the class LM in Eq. (4) are combined using linear interpolation:

$$P_{int}(q|S) = (1-\beta) \cdot P_1(q|S) + \beta \cdot P_C(q|S), \quad (6)$$

where $0 \leq \beta \leq 1$ is an interpolation parameter.

$P_{int}(q|D)$ is calculated in a similar manner. Eq. (3) is then adjusted to give $P_3(q|S)$ as follows:

$$P_3(q|S) = (1-\gamma) \cdot P_{int}(q|S) + \gamma \cdot P_{int}(q|D), \quad (7)$$

where $0 \leq \gamma \leq 1$ is an interpolation parameter.

2.3. Answer extraction

The AE module models the probability of an answer A given a question Q as:

$$P(A|Q) = P(A|W, X), \quad (8)$$

where W is a set of features describing the question-type part of Q, such as *"when"*, *"why"* and *"how"*, etc., while X is a set of features describing the information-bearing part of Q, i.e. what the question is about and what it refers to. For example, in the questions *"Where was the acoustic scene analysis performed?"* and *"When was the acoustic scene analysis performed?"*, the information-bearing parts are identical while the question-type parts differ. Finding the best answer $\hat{A}$ involves a search over all A for the one which maximizes the probability of the above model:

$$\hat{A} = \arg\max_A P(A|W, X). \quad (9)$$

Using Bayes' rule and making various conditional independence and uniform prior distribution assumptions, Eq. (9) can be rearranged to give:

$$\arg\max_A P(A|X) \cdot P(W|A), \quad (10)$$

where $P(A|X)$ is termed the answer retrieval model and $P(W|A)$ the answer filter model. $P(A|X)$ essentially models the proximity of A to features in X. $P(W|A)$ can be viewed as a LM that models the probability of the question-type features W given a candidate answer A.

We will not examine the answer retrieval model and the answer filter model further, see [10] for details.

3. EXPERIMENTAL SETUP

The experimental setup described in this section is very similar, though not identical, to what we used for the actual QAst evaluation.

For our experiments we used the data released for the QAst evaluation task 1 (QA in manual transcripts of lectures) and task 2 (QA in automatic transcripts of lectures). This data contained a development set and an evaluation set (Table 1). The development set consisted of automatic transcripts (ASR) and manual transcripts (MAN) for 10 lectures, a set of questions, and a set of answers for each transcript set. The evaluation set consisted of ASR and MAN for 15 lectures, disjoint from the development set, and a set of questions. All questions were of one of the following answer types: *person*, *location*, *organization*, *language*, *system/method*, *measure*, *time*, *color*, *shape*, and *material*. No answers were provided for the evaluation set, thus we manually extracted them from the transcripts. Word lattices were also available, however, we did not use them. No audio was provided.

Data Set	#Lect.	#Sent.	#Words	WER	#Quest.
Dev. Set	10	2966	54633	32%	45
Eval. Set	15	2917	50986	28%	86

Table 1. Number of lectures, number of sentences, number of words, word error rate and number of questions for each data set after preprocessing.

We cleaned the data by automatically removing fillers and pauses, and performed simple text processing of abbreviations and numerical expressions to ensure consistency between ASR, MAN, questions and answers. ASR was sentence segmented according to the sentence boundaries provided, and MAN was sentence segmented by aligning them with the sentences in ASR. Some of the sentences in ASR were missing in MAN; to ensure consistency, we removed those sentences from ASR. Furthermore, not all questions could be answered based on MAN. The answers to those questions were marked as *"nil"*. Our system is not able to identify whether the answer to a question can be found in the corpus, thus those questions were removed for these experiments. 5 and 14 questions were removed in the development set and the evaluation set, respectively.

For retrieval purposes we filtered out question-type words and stop words (in total 28 words) from the questions. Using the remaining words as query terms, we ranked sentences according to $P_1(q|S)$, $P_2(q|S)$ and $P_3(q|S)$. We optimized weights using the development set and ran evaluations using the evaluation set.

Classes for query expansion were generated based on the overlap in features, which are computed using mutual information, for each word in the vocabulary based on a large text corpus.

Two kinds of experiments were conducted: sentence retrieval and AE. In the case of sentence retrieval, a question was judged to be correctly answered if the highest ranking sentence contained a correct answer, while for answer extraction the exact answer was required. On ASR the correct answer is the words appearing in the same location as the answer for MAN, whether the answer words were correctly recognized or not.

⟨s⟩ SO HERE THE ARTICLES I MEAN THE CORPUS CONSISTS OF BASICALLY NEWS STORIES ⟨/s⟩
⟨s⟩ SO HERE IS A TASK FOR BASICALLY CHINESE TO ENGLISH MACHINE TRANSLATION OF THE USING WHY NEWS CORPUS ⟨/s⟩
⟨s⟩ TO GENERATE A CORPUS AND TO TRAIN ANOTHER N GRAM LANGUAGE MODEL ON THAT PARTICULAR TASK ⟨/s⟩

(a) Top 3 retrieved sentences

CHINESE
ENGLISH
TASK
MACHINE

(b) Top 4 answer candidates

Table 2. Retrieved ASR sentences and extracted answer candidates for the question *"Which language is the Xinhua News Corpus translated to?"*. Correct answer is *"English"*.

Retrieval model	ASR	MAN	Perf. loss
$P_1(q\|S)$	43	53	19%
$P_2(q\|S)$	49	**58**	16%
$P_3(q\|S)$	**51**	**58**	**12%**

Table 3. Number of questions (out of 86) with a correct retrieved sentence in first place, and retrieval performance loss by using automatic transcripts instead of manual transcripts.

	ASR			MAN		
Sentences	P_1	P_2	P_3	P_1	P_2	P_3
Top 1	12	12	12	21	**22**	21
Top 3	10	14	12	19	19	18
Top 5	11	**15**	**15**	19	20	21
Top 10	12	**15**	**15**	18	20	19
All	12			20		

Table 4. Number of questions (out of 86) with a correct extracted answer, supplying the AE module with the highest ranked sentences using $P_1(q|S)$, $P_2(q|S)$ and $P_3(q|S)$, and supplying all sentences.

	ASR		MAN	
Participant	Top 1	Total	Top 1	Total
CLT, Australia	3	13	6	16
DFKI, Germany	9	9	15	19
TokyoTech, Japan	*8*	*18*	*16*	*36*
LIMSI, France	21	29	39	56
UPC, Spain	36	37	52	56

Table 5. QAst results, number of questions (out of 98) with a correct submitted answer, in first place and in total (maximum 5 ranked answers allowed per question). Only the best run for each participant is shown (2 runs per task were allowed.)

4. RESULTS

We evaluated sentence retrieval and AE performance using the setup described in Section 3. All experiments were conducted both on ASR and MAN. Table 2 shows an example of retrieved ASR sentences and extracted answer candidates for the question *"Which language is the Xinhua News Corpus translated to?"*. Here the correct answer, *"English"*, is ranked in second place. Since only the highest ranking answer candidate is evaluated, this question is incorrectly answered. Notice that the highest ranking answer candidate, although incorrect, still is of the correct answer type.

In the sentence retrieval experiments we first investigated the effect of combining the sentence LM and the document LM ($P_2(q|S)$), compared to using only the sentence LM ($P_1(q|S)$). Next, we conducted experiments to examine the effect of our query expansion model by including the class LM in the combination of the sentence LM and the document LM ($P_3(q|S)$). The results are given in Table 3.

In the AE experiments the model presented in Section 2.3 was used. The highest ranking sentences according to the retrieval models were passed to the AE module. We conducted experiments using $P_1(q|S)$, $P_2(q|S)$ and $P_3(q|S)$. On the development set, performance dropped for all experiments when increasing the number of retrieved sentences from five to ten, thus we experimented on the evaluation set with the top 1, 3, 5 and 10 sentences. Next, all sentences in all transcripts were passed directly to the AE module. This is similar to what we have done in previous text based QA evaluations, where we have supplied the AE module with a large amount of data (for example, in TREC we retrieved 500 documents per question from a corpus consisting of over 1 million documents). However, since there were only 15 speech transcripts available in the evaluation set, all documents were retrieved. The topics of the transcripts were similar (speech, image, or signal processing), thus we assumed that all documents supplied were, to some extent, related to the given question. The results of our AE experiments are given in Table 4.

To put the results into perspective, the official results of QAst [2] for us (*TokyoTech*) and the four other participants are given in Table 5. We were the only participant to use an entirely data-driven approach. It should be noted that these results are not directly comparable to the results in Table 4 since a slightly different experimental setup was used. In particular, in QAst we lost several questions due to errors in the preprocessing of numerical expressions. In the experiments described in this paper, wrongly preprocessed numerical answers were accepted if they occurred in the correct location in the token stream. Furthermore, for MAN we improved performance by aligning the text with the sentences in ASR, which we didn't do in QAst.

5. DISCUSSION

5.1. Sentence retrieval

The results of the sentence retrieval experiments show that by retrieving sentences according to $P_2(q|S)$ (combining the sentence LM and the document LM), we are able to get a correct sentence in first place for 57% of the questions when operating on ASR and 67% on MAN. Using $P_2(q|S)$ increases retrieval performance by 14% relative for ASR and 9% relative for MAN, compared to using $P_1(q|S)$ (only the sentence LM). That performance increases more when operating on ASR than on MAN can be explained by word errors in ASR: a sentence will be assigned a much lower probability by the sentence LM if a query term is misrecognized, however, by combining the sentence LM with the document LM, the sentence can still achieve a reasonably high score if the term appears correctly recognized elsewhere in the document.

$P_3(q|S)$ ($P_2(q|S)$ extended with our query expansion model) yields a 4% relative improvement on ASR and no improvement on MAN, compared to using $P_2(q|S)$. The modest improvement on ASR is not significant, given the small amount of data. Manual inspection shows that in some cases our class model based query expansion technique can compensate for word errors when the incorrect word and the correct word are in the same class due to semantic similarities. A typical example is when a singular noun is misrecognized as a plural noun, or vice-versa. We are still experimenting with methods for generating classes more appropriate to the task, for example to take account of phonetic similarity.

It has been shown that spoken document retrieval tasks are able to handle word error rates of 30%-40% with only a small loss in retrieval performance [11]. This is mainly because important words tend to appear more than once in a document, thus chances are high that a query term misrecognized in one location will appear correctly recognized in another location. However, by retrieving sentences instead of documents, such redundancy is less likely. We experience a performance loss of 19% when retrieving sentences in ASR relative to MAN by only using the sentence LM ($P_1(q|S)$), but we are able to reduce performance loss to 12% by combining the sentence LM with the document LM and performing query expansion ($P_3(q|S)$).

5.2. Answer extraction

The results of the AE experiments show that, using the best combinations of LMs and number of sentences experimented with, we are able to extract a correct answer for 17% of the questions in the case of ASR, and for 26% of the questions in the case of MAN. To put our results into perspective, the highest scoring participant in QAst achieved a correct answer for 36% and 52% of 100 questions for ASR and MAN respectively.

Even though there were not many questions in the QAst development and evaluation sets, some trends can be observed. As in the retrieval stage, retrieving sentences by $P_2(q|S)$ results in better AE performance than using $P_1(q|S)$, and again, the improvement is higher on ASR. The improvement is not as large as in the retrieval stage. This was expected, since the AE module only operates on isolated sentences.

Although query expansion was able to slightly increase the retrieval performance on ASR, this did not lead to better AE performance. On MAN, the AE results got worse. Manual inspection shows that for those questions where query expansion has a positive impact on sentence retrieval, the AE module has difficulties taking advantage of this improvement since the AE module is not able to recognize the expanded terms as a query terms.

Supplying the AE model with a small number of sentences, rather than all transcripts, gave better performance for both ASR and MAN. (A beneficial side effect of this improvement was the increase in speed for the AE module, which had to process less data.) In case of ASR, retrieving more than one sentence gave the best result, while on MAN better performance was achieved by supplying only the highest ranking sentence. This can be explained by the difference in sentence retrieval performance: the highest scoring sentence is more likely to contain the correct answer in the case of MAN. When comparing the best results, the AE module is able to extract the correct answer for 17% of the questions in the case of ASR and 26% in the case of MAN, which means AE on ASR perform 32% worse than on MAN.

We analyzed in more detail the AE results on ASR given by $P_2(q|S)$, when the three highest ranked sentences were passed to the AE module. 62 questions have a correct answer (i.e. the words appearing in the same location as the correct answer in the audio) in any of the three supplied sentences. Thus, for this retrieval setup, 62 of 86 questions (72%) is an upper bound on how many correct answers that can potentially be extracted. If the AE module were to extract answers at random from these three sentences, it would on average be able to extract the correct answer for one or two of all 86 questions (2%), given an average of 13 non-stop-words per sentence and assuming that an answer consists of only one word. This represents the lower bound of potential AE performance. Of the 62 questions with a correct answer in the retrieved sentences, 40 have a reference answer that is of the correct answer type. Generally this means that the answer words have been correctly recognized. The AE module gives a correct answer (i.e. an answer equal to the reference answer) for 13 of those 40 questions (33%). Of the questions with a reference answer of the wrong answer type, only 2 of 22 (9%) are answered correctly. Thus answer words of the correct answer type are crucial for good AE performance in our system. This can be explained by the way the answer fil-

ter model (Section 2.3) works: if the answer words in ASR are of the wrong answer type, then $P(W|A)$ will assign a low probability to the correct answer candidate.

The official QAst results showed that we were able to answer questions reasonably well when submitting five answers per question (we ranked third among five participants). However, we were only able to get the correct answer in first place for less than half of the questions where we had a correct answer among the five submitted answer candidates.

6. CONCLUSIONS

In this paper we have presented our language modeling approach to the relatively unexplored field of QA on speech transcripts. We showed that combining sentence LMs with document LMs improves sentence retrieval performance, more so on automatic transcripts than on manual transcripts. Furthermore, retrieving only a few sentences for answer extraction, rather than searching for the answer through all transcripts, produced better QA results. More research on larger data sets is needed to confirm the efficiency of our query expansion model. Generally our sentence retrieval improvements lead to more modest AE improvements. Future research will aim to reduce this mismatch by achieving a tighter coupling between the IR module and the AE module. The effect of extracting answers from lattices, and thereby exploiting confidence scores, will also be examined.

7. ACKNOWLEDGMENTS

This research was supported in part by JSPS and the Japanese government 21st century COE programme.

8. REFERENCES

[1] Voorhees, E. and Tice D., "The TREC-8 Question Answering Track Evaluation", *Proc. TREC-8*, 1999.

[2] Turmo, J., Comas, P., Ayache, C., Mostefa, D., Rosset, S. and Lamel, L., "Overview of QAST 2007", *Working Notes CLEF*, 2007.

[3] Yang, H., Chaisorn, L., Zhao, Y., Neo, S. and Chua, T., "VideoQA: Question Answering on News Video", *Proc. ACM Multimedia*, 2003.

[4] Surdeanu, M., Dominguez-Sal, D. and Comas, P., "Design and Performance Analysis of a Factoid Question Answering System for Spontaneous Speech Transcriptions", *Proc. Interspeech*, 2006.

[5] Harabagiu, S. and Moldovan, D., "Open-Domain Voice-Activated Question Answering", *Proc. COLING*, 2002.

[6] Clarke, C., Cormack, G. and Lynam, T., "Exploiting Redundancy in Question Answering", *Proc. SIGIR*, 2001.

[7] Ponte J. and Croft W. B., "A Language Modeling Approach to Information Retrieval", *Proc. SIGIR*, 1998.

[8] Merkel A. and Klakow D., "Comparing Improved Language Models for Sentence Retrieval in Question Answering", *Proc. CLIN*, 2007.

[9] Sun, R., Ong C. and Chua T., "Mining Dependency Relations for Query Expansion in Passage Retrieval", *Proc. SIGIR*, 2006.

[10] Whittaker, E., Novak, J., Chatain, P. and Furui, S., "TREC 2006 Question Answering Experiments at Tokyo Institute of Technology", *Proc. TREC-15*, 2006.

[11] Garofolo, J., Auzanne, G. and Voorhees, E, "The TREC Spoken Document Retrieval Track: A Success Story", *Proc. TREC-8*, 1999.