Language Modeling Approach for Retrieving Passages in Lecture Audio Data

Koichiro Honda, Tomoyosi Akiba

Department of Information and Computer Sciences, Toyohashi University of Technology,
1-1 Hibarigaoka, Tenpaku-cho, Toyohashi, JAPAN
{honda, akiba}@cl.ics.tut.ac.jp

Abstract

Spoken Document Retrieval (SDR) is a promising technology for enhancing the utility of spoken materials. After the spoken documents have been transcribed by using a Large Vocabulary Continuous Speech Recognition (LVCSR) decoder, a text-based ad hoc retrieval method can be applied directly to the transcribed documents. However, recognition errors will significantly degrade the retrieval performance. To address this problem, we have previously proposed a method that aimed to fill the gap between automatically transcribed text and correctly transcribed text by using a statistical translation technique. In this paper, we extend the method by (1) using neighboring context to index the target passage, and (2) applying a language modeling approach for document retrieval. Our experimental evaluation shows that context information can improve retrieval performance, and that the language modeling approach is effective in incorporating context information into the proposed SDR method, which uses a translation model.

1. Introduction

Traditionally, human beings have used spoken language mainly for communication. However, advances in speech recognition technologies will make it possible to use spoken language in addition to written language as a medium for storing and transmitting knowledge. In practice, audio data such as broadcast news, lectures, and Weblog-style recording in podcasts is increasingly available via the Internet. However, these audio data sources are difficult to reuse because efficient searching within them is much more difficult than for textual material.

Spoken Document Retrieval (SDR) is a promising technology for solving these problems. It was extensively evaluated in the Text Retrieval Conference (TREC) SDR Tracks (Garofolo et al., 1999). In the first (TREC-6) SDR Track, the set task was *known-item retrieval*, in which the system was to find a particular, half-remembered term in a document collection. In the later TREC-7 to TREC-9 SDR Tracks, *ad hoc retrieval* was investigated, in which the system was required to return ranked, relevant documents that were similar for given topics. In these TREC SDRs, a simple method that used N-best automatic transcriptions, obtained by using an LVCSR system for indexing spoken documents, was investigated and shown to be effective.

Since then, SDR research has moved to more difficult tasks in high Word-Error-Rate (WER) conditions. In contrast to the older TREC SDR Tracks, much recent work (cheng Pan et al., 2007; Shao et al., 2008) has focused on traditional known-item retrieval. For example, spoken term detection, aiming at a rapid and accurate search for a specified term, has been evaluated by the National Institute of Standards and Technology [1] . However, ad hoc retrieval still seems to be more practical for the tasks required in SDR. In practice, ad hoc retrieval that targets textual material is still being studied actively in the research fields of Information Retrieval (IR) and natural language processing.

The most straightforward method for ad hoc SDR is simply to use automatic transcriptions of the target spoken documents for indexing, as has been tried in the TREC SDR Tracks. After transcription via an LVCSR system, a text-based ad hoc retrieval method can simply be applied to the transcribed documents. However, recognition errors will significantly degrade the IR performance. In particular, because words that are Out Of Vocabulary (OOV) for the recognition dictionary of the LVCSR decoder do not appear in the transcribed text, a query constructed from such words will never match any document in the target collection.

To address this problem, we have previously proposed a method that can fill the gap between automatically transcribed text and correctly transcribed text by using a statistical translation technique (Akiba and Yokota, 2008). In this paper, we extend the method by (1) using neighboring context to index the target passage, and (2) applying a language modeling approach for document retrieval. Our experimental evaluation shows that the context information can improve retrieval performance, and that the language modeling approach is effective in incorporating context information into the proposed SDR method, which uses a translation model.

The remainder of this paper is organized as follows. Section 2. describes the training and test data used in this work. Section 3. explains our previously proposed ad hoc retrieval method using a translation model. Section 4. introduces the proposed extensions. In Section 5., we evaluate the proposed method by comparison with conventional document retrieval methods. Finally, we conclude and describe future work in Section 6..

2. Data

2.1. Test Collection for SDR

A test collection for text document retrieval comprises three elements: a document collection in a target domain, a set of queries, and the results of relevance judgments, i.e. sets of relevant documents that are selected from the collection for each query in the query set. For SDR, two additional elements are necessary, namely the manual and automatic transcriptions of the spoken document collection.

We used the CSJ test collection (Akiba et al., 2008) (Akiba et al., 2009) both for training our retrieval model and for evaluating it. The target document collection is 2,702

[1] http://www.nist.gov/speech/tests/std/

lectures selected from the Corpus of Spontaneous Japanese (Maekawa et al., 2000). This amounts to more than 600 hours of speech, which is comparable to the TREC SDR test collection (Garofolo et al., 1999). Together with the speech data items themselves, their manual transcriptions are included in the CSJ.

The test collection contains 39 queries about information described in part of a lecture. The relevance of such queries is judged against segments of varying length from the lectures, called *passages*. Relevant passages are assigned to one of two classes, "relevant" or "partially relevant", according to their degree of relevance.

The test collection also includes the automatic transcriptions obtained via a Japanese LVCSR decoder. The WER was about 20%, which is comparable with that for the TREC SDR task (Garofolo et al., 1999).

Table 1 gives a summary of the CSJ test collection compared with the TREC-9 SDR test collection.

2.2. Retrieval Task Definition

The primary retrieval task specified for the test collection is somewhat different from a conventional retrieval task, where the target unit of retrieval is predefined and fixed, such as an article in a newspaper. Therefore, we chose to redefine the conventional retrieval task, instead of specifically searching for variable length segments in the collection.

First, we created *pseudopassages* by automatically dividing each lecture into a sequence of segments, with N utterances per segment. At 15 utterances per segment, there are 60,202 pseudopassages, and the number of words per document averages 102.1.

Next, we assigned a relevance label to the retrieved pseudopassages as follows: if the pseudopassage shares at least one utterance with the relevant passage specified in the "golden file", then the pseudopassage is labeled as "relevant". Two degrees of relevance were used in the evaluation as follows:

R The passages labeled "relevant" are used to decide the relevant pseudopassages.

R+P The passages labeled either "relevant" or "partially relevant" are used to decide the relevant pseudopassages.

3. SDR Using Word Translation Model

After using an LVCSR decoder to obtain transcriptions automatically, a conventional text-based document retrieval method can be applied to index the transcribed text documents. However, one of the most common problems arises from recognition errors. In particular, words that are OOV for the LVCSR decoder can never be utilized as indices in text-based document retrieval. Such indexing errors seriously degrade the IR performance.

As an alternative, Our previously proposed SDR method (Akiba and Yokota, 2008) estimates the correct transcriptions from the automatic transcriptions and uses them to generate the indices. For the estimation, we use a word translation model inspired by statistical machine translation. The word translation model gives the probability $t(f|e)$ that a word f appears in the correct transcription, given a word e in the automatic transcription.

3.1. Estimation of the Word Translation Model

To estimate the word translation probability $t(f|e)$, we use the parallel text that comprises pairs of automatically and manually transcribed sentences.

First, both the automatic and manual transcriptions in the parallel text are morphologically analyzed and segmented into word sequences. Second, the pairs of word sequences are word aligned by using dynamic-programming matching guided by an edit-distance function. In the resulting word alignments, the exactly matched word pairs are retained and the remaining unmatched words are fractionally realigned to avoid crossing the former matching alignments.

Here, we use the *Simple Distribution Method* described in (Akiba and Yokota, 2008) for the realignment, where each unmatched word in the automatic transcription is fractionally and uniformly aligned with every unmatched word in the manual transcription to avoid crossing the fixed matched-word alignments. For example, given word sequences $\cdots e_p e_{p+1} \cdots e_{p+l} e_{p+l+1} \cdots$ from the automatic transcription and $\cdots f_q f_{q+1} \cdots f_{q+m} f_{q+m+1} \cdots$ from the manual transcription, suppose the word pairs (e_p, f_q) and (e_{p+l+1}, f_{q+m+1}) are exactly matched but the word sequences between them are not matched. Then, the (fractional) count $tc(e, f)$ of a word alignment between e and f is

$$\begin{aligned} &tc(e_p, f_q) = 1, \\ &tc(e_i, f_j) = \frac{1}{m} \qquad (p < i \leq p+l, q < j \leq q+m), \\ &tc(e_{p+l+1}, f_{q+m+1}) = 1. \end{aligned}$$

Finally, the resulting (fractional) word alignments are summed over all the parallel text to obtain the fractional counts. From them, the word translation probabilities $t(f|e)$ are obtained by maximum likelihood estimation.

3.2. Spoken Document Indexing Using the Word Translation Model

The estimated word translation model $t(f|e)$ is used to calculate the expected term frequency $TF_f(f, D)$ for the word f, which should appear in the manual transcription of the spoken document D. This is estimated by

$$TF_f(f, D) = \sum_{e \in E_D} t(f|e) TF_e(e, D). \qquad (1)$$

For smoothing purposes, it is also interpolated with the original term frequency

$$\tilde{TF}_f(f, D) = \lambda E(TF_f(f, D)) + (1 - \lambda) TF_e(f, D). \quad (2)$$

where $TF_e(e, D)$ is the term frequency of the word e observed in the automatic transcription of D. In this paper, λ is fixed at 0.5.

A threshold α is introduced to avoid using low-frequency words in the indexing. Because the expected term frequency $\tilde{TF}_f(f, D)$ is consistent with the $TF_e(e, D)$ that is calculated from the statistics of D, the conventional vector

Table 1: A comparison between the TREC-9 SDR and the CSJ SDR test collections.

	TREC-9 SDR	CSJ SDR
Language	English	Japanese
Target documents	Broadcast news	Lecture speech
Quantity	557 hours	623.6 hours
Documents	21,754	2,702 (60,202 seg.*)
Words per document	169	2324.9 (102.1 per seg.*)
Queries	50	39
Reference Transcription	closed caption (WER 10.3%)	manual transcription
WER	26.7%	21.4%

* A sequence of 15 utterances is considered a segment.

space IR model based on Term Frequency–Inverse Document Frequency (TF–IDF) term weighting can be used for document retrieval. We used *GETA* [2] as the IR engine in our SDR system.

4. Proposed Methods

4.1. Using the Neighboring Context to Index the Target Passage

Our retrieval task is to find relevant pseudopassages in lectures. Pseudopassages from the same lecture may be related to each other in our task, whereas the target documents are considered to be independent of each other in a conventional document retrieval task. In particular, the neighboring context of a target pseudopassage should contain related information, because we automatically divide a lecture into fixed-length pseudopassages without considering its content. It would seem appropriate for our retrieval task to use the neighboring context to index the target pseudopassage. A similar method was applied in TREC SDR TRACK (Johnson et al., 1999).

Normally, a document (pseudopassage) D is indexed by its own term frequencies $TF(t, D)$ of the terms $t \in D$. This can be extended to use the neighboring context for indexing. For the context $context_n(D)$, the preceding n utterances and the following n utterances are used. Therefore, we use

$$TF_{ext}(t, D) = \beta TF(t, D) + TF(t, context_n(D)), \quad (3)$$

where β is introduced to specify the relative importance of D and $context_n(D)$.

We now combine this document expansion method with the word translation model. This means using $\tilde{TF}_F(t, D)$ and $\tilde{TF}_F(t, context_n(D))$ obtained from Equation (2), instead of $TF(t, D)$ and $TF(t, context_n(D))$, to obtain

$$\tilde{TF}_{ext}(t, D) = \beta \tilde{TF}(t, D) + \tilde{TF}(t, context_n(D)). \quad (4)$$

4.2. Applying the Language Modeling Approach for SDR

Recently, the effectiveness of the language modeling approach for IR has been reported (Croft and Lafferty, 2003). Its probabilistic framework seems to match our translation model better than the traditional vector space retrieval model. Here, for document reranking, we use the probability $P(Q|D)$ that a query Q is constructed from a relevant document D

$$P(Q|D) = \prod_{q \in Q} P(q|D). \quad (5)$$

$P(q|D)$ is estimated by

$$P(q|D) = (1-\gamma)\frac{TF(q, D)}{\sum_t TF(t, D)} + \gamma \frac{TF(q)}{\sum_t TF(t)}, \quad (6)$$

where $TF(q)$ is the global term frequency of a query term q calculated from the target document collection C by

$$TF(q) = \sum_{D \in C} TF(q, D). \quad (7)$$

The language model can be combined directly with the word translation model

$$P(q|D) = \sum_e P(q|e, D)P(e|D) \approx \sum_e t(q|e)P(e|D). \quad (8)$$

$$\begin{aligned} P_{tmodel}(q|D) &= (1-\mu)\sum_e P(q|e, D)P(e|D) + \mu P(q|D) \\ &\approx (1-\mu)\sum_e t(q|e)P(e|D) + \mu P(q|D). \end{aligned} \quad (9)$$

5. Evaluation

5.1. Evaluation Metric

We used 11-point Average Precision (AP) as our evaluation metric. , which is obtained by averaging precisions as follows:

$$\begin{aligned} IP(x) &= \max_{x \le R_i} P_i, \\ AP &= \frac{1}{11}\sum_{i=0}^{10} IP(\frac{i}{10}), \end{aligned}$$

where R_i and P_i are the recall and the precision, respectively, up to the i-th retrieved document. In practice, we retrieved 1000 documents for each query in calculating the AP.

[2] http://geta.ex.nii.ac.jp

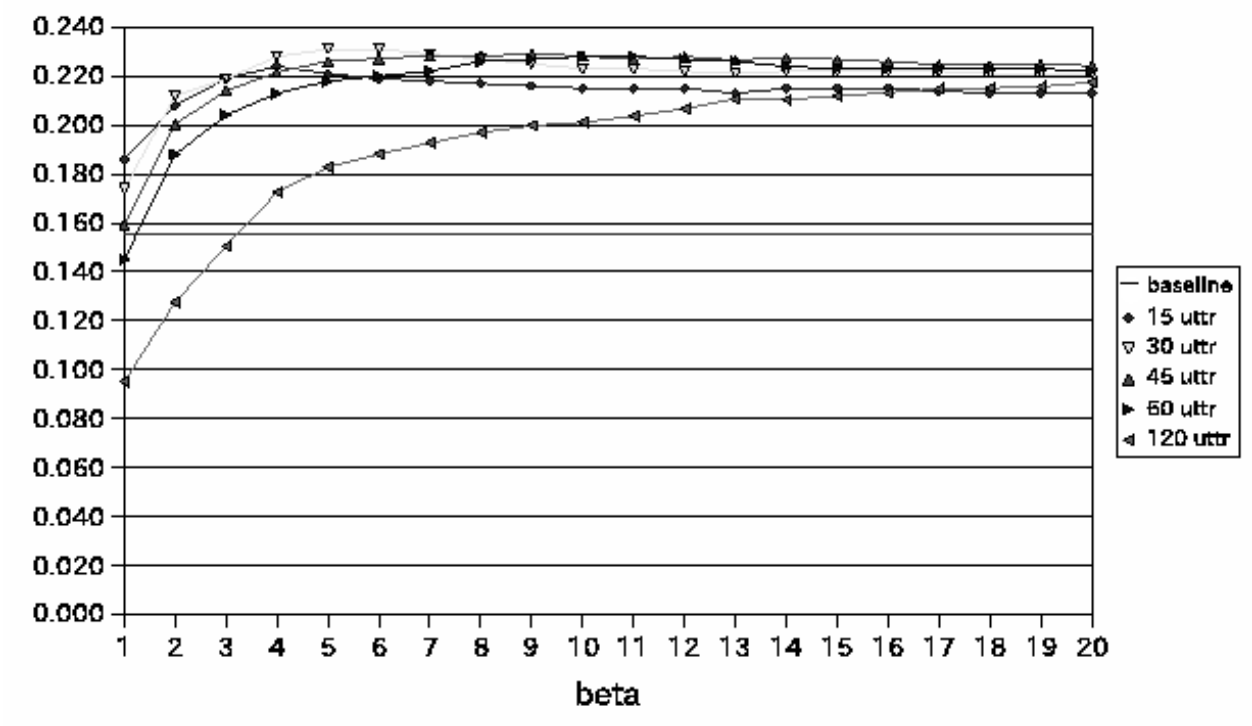

Figure 1: APs using neighboring context.

5.2. Translation Model Training

The paired manual and automatic transcriptions contained in the CSJ test collection described were used as the parallel text for training the translation model. Because they were also the target documents of the test collection, we applied the cross-validation style indexing scheme; the nine tehnth of the target documents were used to train the translation model, which was then used to index the rest one tenth of the target documents, and the process was reported ten-fold. we applied the following training scheme, inspired by the cross-validation used in statistical testing, to make the evaluation settings available to the test data.

First, the target documents (lectures) were randomly divided into 10 groups. Next, nine of these groups were used to train the translation model, and the resulting model was then used to index the documents in the remaining group. This process was repeated tenfold to index all documents in the test collection.

5.3. Results

5.3.1. Baseline

As our baseline, only those terms appearing in the target document (pseudopassage) were used for indexing. We compared three representations for the documents, namely the 1-best automatically transcribed text, the union of the 10-best automatically transcribed texts, and the reference manually transcribed text. The transcribed texts were morphologically analyzed and segmented into words for indexing. The vector space model with TF–IDF term weighting was used as the retrieval method with pivoted normalization

The 11-point APs were 0.155, 0.177, and 0.180 for 1-best, 10-best, and manual transcriptions, respectively.

5.3.2. Using Neighboring Context

For indexing, a document (pseudopassage) was extended by using its context, as described in Section 4.1.. The context size n, referring to the use of both the preceding n and following n utterances, was set to one of 15, 30, 45, 60, or 120, and the weighting parameter was set to a value in the range 1 to 20. The traditional vector space model with TF–IDF term weighting was used as the retrieval method. Only the 1-best recognition candidate was used for indexing. Figure 1 gives the results, which show that the neighboring context is useful for indexing pseudopassages.

5.3.3. Combining with the Word Translation Model

Next, we used both the word translation model and the neighboring context together. For this, we set $n = 15$ and $\beta = 3$. The vector space model was again used as the retrieval model. We compared the **baseline** (indexed using only its own document), the method using the word translation model (referred to as **trans**), the method using the context (**context**), and that the combined method (**trans+context**). We also compared these methods with the results on manual transcriptions instead of automatic transcriptions, i.e. the **baseline** method (**baseline(reference)**) and the **context** method (**context(reference)**). Figure 2 and 3 show the results using R degree of relevance. The bars on the left side for each item in Figure 2 and 3 show the results. These results show that using either the word translation model alone or the neighboring context alone improved the retrieval performance. However, combining the word translation model with the neighboring context degraded the performance. We investigated the results in detail, seeking the reason why the combination did not work well, and found that the joint use of the two methods considerably increased the document frequency values for the terms in the lectures

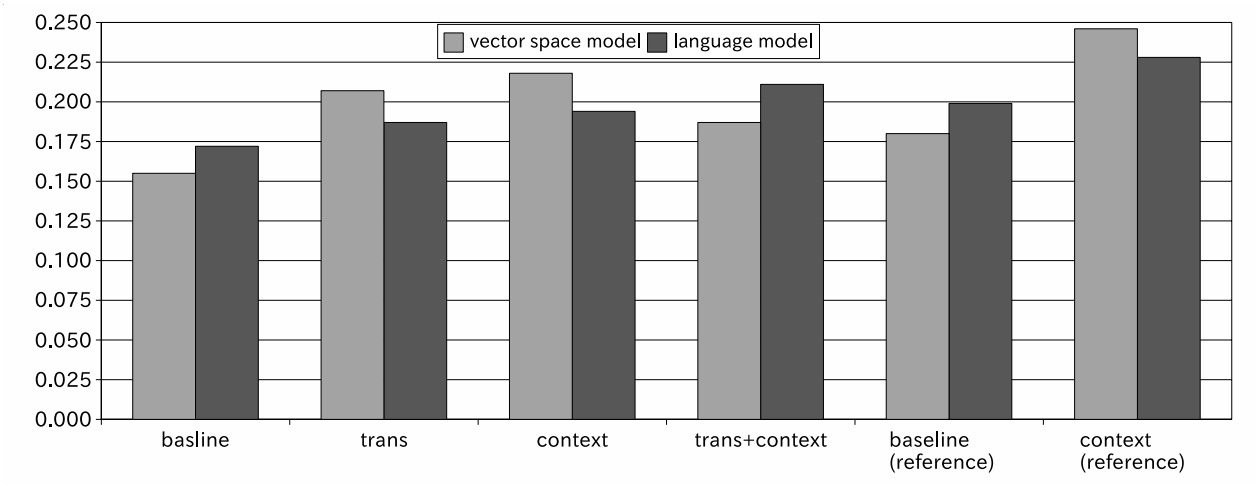

Figure 2: APs for all compared methods using R degfree of relevance.

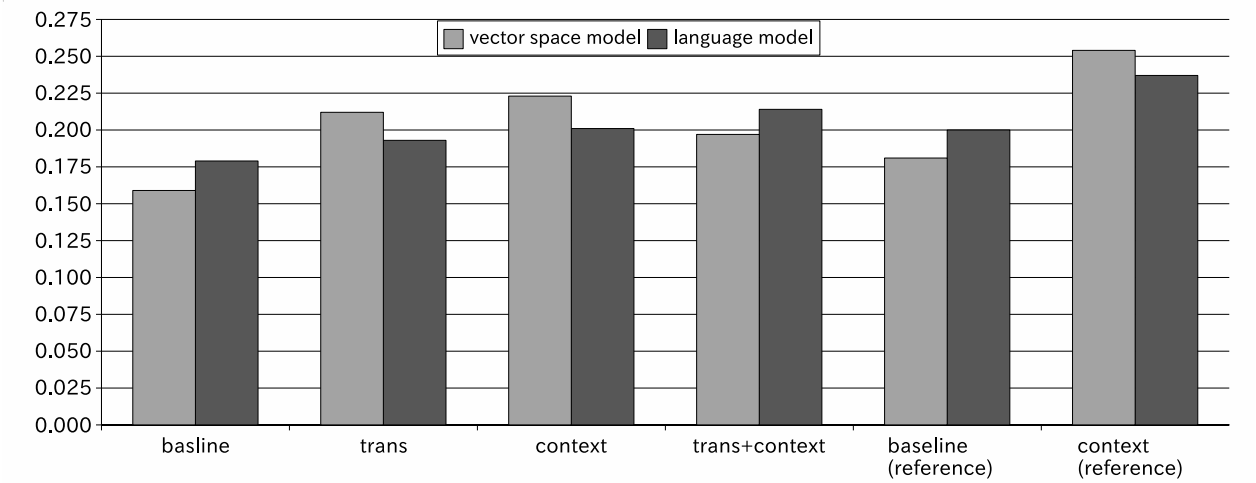

Figure 3: APs for all compared methods using R+P degree of relevance.

as a whole, so that the importance of a term could not be measured appropriately. Additionaly the combination has increased the noisy frequency.

5.3.4. Language Modeling Approach

Instead of using the vector space model with TF–IDF term weighting, we applied the language modeling approach for our retrieval model, as described in Section 4.2.. However, for combining with the word translation model, we introduced an approximate calculation for $P(Q|D)$ to simplify the implementation. Specifically, instead of Equation (9), $P(q|D)$ was calculated via the expected TF:

$$P(q|D) = (1-\mu)\frac{\tilde{TF}(q,D)}{\sum_t \tilde{TF}(t,D)} + \mu\frac{\tilde{TF}(q)}{\sum_t \tilde{TF}(t)}, \quad (10)$$

In order to reduce the computation time, the model was implemented as follows. Firstly, top ranked 1000 documents were retrieved by using the simplified language model given as follows:

$$P(q|D) = (1-\gamma)\frac{\tilde{TF}(q,D)}{\sum_t \tilde{TF}(t,D)} + \gamma\frac{\tilde{TF}(q)}{\sum_t \tilde{TF}(t)}, \quad (11)$$

where $\tilde{TF}(q,D)$ was obtained from either Equation (2) or Equation (4), depending on whether the neighboring context was used or not used, respectively, and $\tilde{TF}(q)$ was obtained as follows:

$$\tilde{TF}(q) = \sum_D \tilde{TF}(q,D). \quad (12)$$

Then they were reranked by using the Equation (9).

The bars on the right side for each item in Figure 2 and 3 show the results. Again, using either the word translation model or the neighboring context improves the performance even with the language modeling approach. The results also show that the joint use of both methods further improves the performance.

However, among all the methods, language modeling did not perform the best. It did not outperform the best result obtained by the vector space model extended by using the neighboring context. This may be caused partly by our current implementation and by our choice of formulation from among the language modeling approaches.

6. Conclusions

In this paper, we extended our SDR method that uses a word translation model in two ways. Firstly, we extended the target passages by including their neighboring context. Secondly, we applied a language modeling approach for document retrieval instead of the traditional vector space retrieval model. Our experimental evaluation shows that context information can improve retrieval performance, and that the language modeling approach is effective in incorporating context information into the proposed SDR method that uses a translation model.

In future work, we would like to apply a better language modeling method and a better implementation for document retrieval, to improve the overall performance. We will also investigate the retrieval of variable length passages directly without segmenting lectures into fixed length pseudopassages beforehand.

7. References

Tomoyosi Akiba and Yusuke Yokota. 2008. Spoken document retrieval by translating recognition candidates into correct transcriptions. In *Proceedings of International Conference on Speech Communication and Technology*, pages 2166–2169.

Tomoyosi Akiba, Kiyoaki Aikawa, yoshiaki Itoh, Tatsuya Kawahara, Hiroaki Nanjo, Hiromitsu Nishizaki, Norihito Yasuda, Yoichi Yamashita, and Katunobu Itou. 2008. Test collections for spoken document retrieval from lecture audio data. In *Proceedings of International Conference on Language Resources and Evaluation*.

Tomoyosi Akiba, Kiyoaki Aikawa, Yoshiki Itoh, Tatsuya Kawahara, Hiroaki Nanjo, Hiromitsu Nishizaki, Norihito Yasuda, and Yoichi Yamashita Katunobu Itou. 2009. Developing an SDR test collection from Japanese lecture audio data. In *APSIPA ASC 2009*, pages 324–330.

Yi cheng Pan, Hung lin Chang, Berlin Chen, and Lin shan Lee. 2007. Subword-based position specific posterior lattices (S-PSPL) for indexing speech information. In *Proceedings of International Conference on Speech Communication and Technology*, pages 318–321.

W. Bruce Croft and John Lafferty, editors. 2003. *Language Modeling for Information Retrieval*. Kluwer Academic Publishers.

John S. Garofolo, Cedric G. P. Auzanne, and Ellen M. Voorhees. 1999. The TREC spoken document retrieval track: A success story. In *Proceedings of TREC-9*, pages 107–129.

S.E. Johnson, P. Jourlin, K.sparck Jones, and P.C. Woodland. 1999. Spoken document retrieval for TREC-9 at cambridge university. In *Proceedings of TREC-9*.

Kikuo Maekawa, Hanae Koiso, Sadaoki Furui, and Hitoshi Isahara. 2000. Spontaneous speech corpus of Japanese. In *Proceedings of LREC*, pages 947–952.

Jian Shao, Qingwei Zhao, Pengyuan Zhang, Zhaojie Liu, and Yonghong Yan. 2008. A fast fuzzy keyword spotting algorithm based on syllable confusion network. In *Proceedings of International Conference on Speech Communication and Technology*, pages 2405–2408.