Spoken document retrieval using both word-based and syllable-based document spaces with latent semantic indexing

Ken Ichikawa*, Satoru Tsuge†, Norihide Kitaoka*, Kazuya Takeda* and Kenji Kita‡
*Nagoya University,†Daido University,‡The University of Tokushima
E-mail: ichikawa.ken@g.sp.m.is.nagoya-u.ac.jp

***Abstract*—In this paper, we propose a spoken document retrieval method using vector space models in multiple document spaces. First we construct multiple document vector spaces, one of which is based on continuous-word speech recognition results and the other on continuous-syllable speech recognition results. Query expansion is also applied to the word-based document space. We proposed to apply latent semantic indexing (LSI) not only to the word-based space but also to the syllable-based space, to reduce dimensionality of the spaces using implicitly defined semantics. Finally, we combine the distances and compare the distance between the query and the available documents in various spaces to rank the documents. In this procedure, we propose to model the document by hyperplane. To evaluate our proposed method, we conducted spoken document retrieval experiments using the NTCIR-9 SpokenDoc data set. The results showed that using the combination of the distances, and using LSI on the syllable-based document space, improved retrieval performance.**

I. Introduction

There are many kinds of media data, such as pictures, movies, music, speech, and so on, on the Internet, and opportunities to retrieve such data are increasing. In this paper, we focus on the speech data, which are contained in most of the media data. We call such speech data "spoken documents". Typical information retrieval methods for speech data first transform the target speech data into word or sub-word sequences using an automatic speech recognizer. By using a text retrieval technique, the target information is retrieved from the spoken documents. Spoken document retrieval methods have become an essential technique for information retrieval method.

In this paper, we propose a novel spoken document retrieval method in which the target documents and queries are represented as vectors by a vector space model (VSM). The VSM maps the target documents and queries into a vector space associated with the index terms.

Generally, the vectors mapped by VSM are high-dimensional and sparse. Hence, they are sensitive to noise, and it is also difficult to capture the underlying semantic structure. The latent semantic indexing (LSI) technique is one way to overcome these problems [4][5]. The proposed method applies LSI to the vector space for spoken document retrieval. Because there are some recognition errors and out-of-vocabulary (OOV) terms in spoken documents which are transcribed by speech recognizers, it is difficult to retrieve documents using only traditional text retrieval methods. Hence, some additional methods have been proposed for spoken document retrieval. One method is to use sub-word units instead of words for the indices [1]. This method avoids the OOV problem. In addition, a method which combines phoneme-based recognition results with word-based recognition results has been proposed for spoken term detection [2]. To deal with recognition errors, indices have been constructed using words in a lattice/confusion matrix[3].

In addition, we use the continuous word recognition results and the syllable recognition results for the target documents in our method. Hence, the proposed method maps a target document to multiple vector spaces because it is possible to choose the word- and syllable-based index terms for constructing vector spaces. Because the proposed method represents a document in different ways in multiple spaces, we can calculate the distance between the document vector and the query vector in each space, because each vector space represents different information about the documents. If these different kinds of information are combined efficiently, retrieval performance can be improved. Therefore, in this paper we propose a distance combination method for combining the information from each vector space.

On the other hand, it is difficult to retrieve relevant documents using the query vector if the number of index terms in the query is small. To increase the number of index terms in the query, query expansion methods have been proposed [6]. Query expansion methods often collect related words from the Internet and construct a new query vector using the original query vector and the collected word vector (which we call "expansion vector") by linear combination. In this paper, we propose a method to model the targets as a hyperplane, in which the weight parameter is automatically determined.

II. Proposed Spoken Document Retrieval Method

A. Multiple Vector Space

VSM maps the target documents and query to vector spaces constructed using the index terms. In this paper, we use multiple kinds of index terms, (both word-based and syllable-based index terms) because our target documents are speech recognition results and thus they are noisy because of speech recognition errors. Multiple levels of index terms are expected to give complementary effects on the retrieval. Our proposed method uses two kinds of term weighting methods for index term weighting:

- TF-IDF Weight
  First, we calculate TF (Term Frequency) using the N-best speech recognition candidates:

$$d_{ij} = \sum_{n=1}^{N} \frac{T_{ij}^n}{n}, \tag{1}$$

  where d_{ij} indicates the TF value of the jth index term in the ith document. T_{ij}^n is the term frequency of the jth index term in nth speech recognition candidate of the ith document. Then we calculate TF-IDF using the following equation:

$$d_{ij} = \sum_{n=1}^{N} \frac{T_{ij}^n}{n} I_j, \quad I_j = \log\left(\frac{D}{D_j}\right), \tag{2}$$

where I_j indicates the value of IDF (Inverse Document Frequency) for jth index term. D and D_j are the total number of documents and the number of documents in which jth index term appears, respectively.

- Binary Weight
  With this weighting method, the term weight is 1 when the term appears in the document. If the term does not appear in the document, the term weight is 0.

By using these index term weights for constructing the vector spaces, the target document can be mapped on multiple vector spaces using VSM.

B. Latent Semantic Indexing in syllable-based vector space

The target document is represented as a vector in the vector space consisted of index terms. Target document vectors are high-dimensional and sparse because the number of index terms is large, therefore they are sensitive to noise, and it is also difficult to capture the underlying semantic structure. So our proposed method applies LSI to the vector spaces to overcome these problems. LSI finds a latent semantic space (a concept space) of low-dimension by performing singular value decomposition on the original vector space and then retrieves documents in the latent semantic space [4][5]. In this paper, we propose to apply LSI not to word-based document space, but to syllable-based space. Syllable sequence itself does not have meanings, but the LSI is expected to distill the implicit semantics from the co-occurrence of the sequences.

C. Combination of distances in multiple space

The proposed method maps a target document to multiple vector spaces by using various index terms and term weighting. It is then possible to calculate the distance between each vectors in each space and use all the information for retrieving documents. The proposed method combines the distances calculated in the multiple vector spaces, using the following formula:

$$S = 1 - \sum_k \beta_k s_k(\boldsymbol{d}_i, \boldsymbol{q}), \tag{3}$$

where $s_k()$ and S indicate the cosine similarity calculated in vector space k and the combined distance for retrieval, respectively. β_k is the weight parameter of each cosine similarity. Using the distance calculated by formula (3), the proposed method ranks and retrieves documents. We expect to improve retrieval performance by using a combination of information from various spaces rather than from a single space.

D. Modelization by hyperplane of Query Expansion

Generally, it is difficult to retrieve relevant documents if the number of index terms in the query is small. Therefore, query expansion is often used in order to increase the number of index terms in the query. To expand the query, first, our system collects web pages from the Internet using keywords (in our case, nouns) in the query. Then, using the collected web pages, an expanded vector is constructed in the same manner as the document vector. Expanded query vector $\hat{\boldsymbol{q}}$ used for retrieval is calculated using the following equation:

$$\hat{\boldsymbol{q}} = (1-\alpha)\boldsymbol{q}_o + \alpha \boldsymbol{q}_e, \tag{4}$$

where $\boldsymbol{q}_o$ and $\boldsymbol{q}_e$ are the original query vector constructed from the query sentence, and the expansion vector constructed from the collected web pages, respectively. α is the weight parameter between the original query vector and the expansion vector. Usually, the value used for α fixed a prior for all the target documents. This means that the typical target document is modeled by a vector which is a linear sum of $\boldsymbol{q}_o$ and $\boldsymbol{q}_e$ using fixed weight. This models however, seems to be too constrained. To relax the constraint, we propose to model the typical target document set as a hyperplane spanned by $\boldsymbol{q}_o$ and $\boldsymbol{q}_e$. Using this model, we evaluate the relevance of document vector $\boldsymbol{d}_i$ by measuring the angle between $\boldsymbol{d}_i$ and the hyperplane. This approach is illustrated in Fig. 1. Therefore, the optimal weight of parameter α is calculated using following equation:

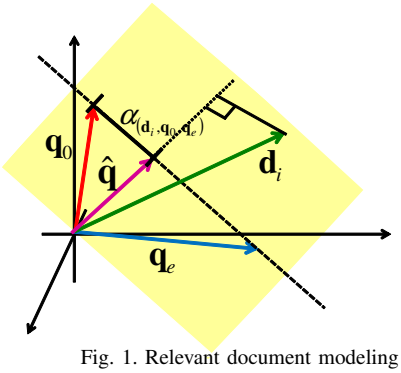

Fig. 1. Relevant document modeling

$$\begin{aligned}
\alpha(\boldsymbol{d}_i, \boldsymbol{q}_o, \boldsymbol{q}_e) &= argmax_\alpha(s(\boldsymbol{d}_i, \hat{\boldsymbol{q}})) \\
&= argmax_\alpha \left(\frac{\boldsymbol{d}_i \cdot \hat{\boldsymbol{q}}}{|\boldsymbol{d}_i||\hat{\boldsymbol{q}}|} \right) \\
&= argmax_\alpha \left(\frac{\boldsymbol{d}_i \cdot ((1-\alpha)\boldsymbol{q}_o + \alpha \boldsymbol{q}_e)}{|\boldsymbol{d}_i||((1-\alpha\boldsymbol{q}_o) + \alpha \boldsymbol{q}_e)|} \right).
\end{aligned} \tag{5}$$

where $s()$ indicates the cosine similarity. $\alpha(\boldsymbol{d}_i, \boldsymbol{q}_o, \boldsymbol{q}_e)$ can be calculated analytically.

III. EXPERIMENT

In order to evaluate the proposed method, we conducted spoken document retrieval experiments under the conditions of the SpokenDoc task of NTCIR-9 [7]. First, using the development set which was called "Dry Run" in NTCIR-9, we evaluated the proposed method and tuned its parameters.

Then we evaluated the proposed method by using the open test set called "Formal Run" in NTCIR-9, which is described in Section III-C.

A. Experimental conditions

For target documents, we used the speech recognition results of lecture speech, which consist of Japanese oral presentations in the CSJ database [8] provided by the NTCIR-9 organizers. These recognition results consist of continuous word recognition results and continuous syllable recognition results. There were 2,702 target documents, and the accuracy of word and syllable recognition were about 70% and 75%, respectively.

As index terms for word-based VSM, we chose the words whose morphemes are nouns, alphabet sequences, and katakana sequences, from a vocabulary list used in a continuous word recognizer. The number of these index terms was 14,716. As index terms for syllable-based VSM, we used syllable 3-grams which were selected from continuous syllable recognition results. The number of these index terms was 169,363. Binary weighting and TF-IDF were used as the index term weighting methods for word-based VSM, and TF-IDF was used as the index term weighting method for syllable-based VSM.

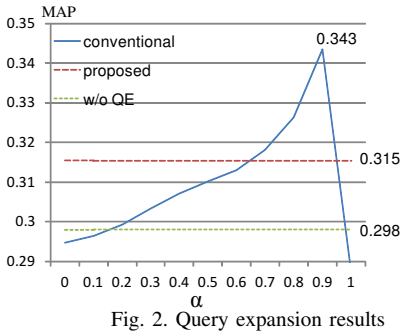

Fig. 2. Query expansion results

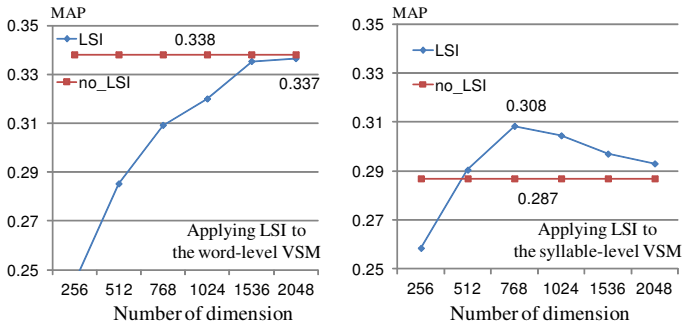

Fig. 3. LSI results

For syllable-based VSM, the queries were translated into Japanese katakana sequences. For query expansion, using the nouns from each query, we obtained web pages using Google's search-engine. The index terms selected from these web pages, using Google JSON/Atom Custom Search API [9], were used for calculating the expanding vector ($\boldsymbol{q}_e$ in formula (4)). We used Mean Average Precision (MAP) as our measure for evaluation [7].

B. Results of Dry Run

Before finalizing our proposed method, we used the NTCIR-9 "Dry Run" test set for evaluation, and this section describes our experimental results. In the experiments, we used 39 Dry Run queries. Each query consisted of one sentence. We describe our results in the following subsection:

- Effectiveness of query expansion (III-B1),
- Evaluation of LSI in word-based and the syllable-based vector spaces (III-B2),
- Evaluation of distance combinations (III-B3).

1) Effectiveness of query expansion: In this section, we describe the experimental results of query expansion. Query expansion was applied to word-based VSM with TF-IDF weighting, and this method achieved the highest retrieval score of the various methods we evaluated. Fig. 2 shows the MAP depending on the weight parameter of query expansion, α. The number of speech recognition candidates and the number of web pages for query expansion used in the proposed and conventional methods, was 2 and 21, and 3 and 21, respectively . These parameters showed the highest scores in our preliminary experiment. Fig. 2 shows that both query expansion methods, conventional and proposed, outperformed w/o QE which was the highest score using TF-IDF wight of word-based VSM without query expansion. Therefore, query expansion was shown to be a useful method for spoken document retrieval.

Comparing the conventional query expansion method and the proposed one, the proposed method did not achieve the MAP of the conventional method. The proposed method determines the weight parameter of query expansion where the cosine distance between the document vector and the expanded query vector is the smallest. When we investigated the details of the experimental results, most of weight parameters calculated by the proposed method had values from 0.9 to 1.1. In this experiment, if the weight parameter was greater than 1.0, the weight parameter was set to 1.0 and if the weight parameter was less than 0.0, the weight parameter was set to 0.0. We believe using an expanded query vector whose weight parameter was greater than 1.0 affected the retrieval results, and that this caused the degradation of the MAP score. We will investigate the details of this experimental result in the future.

2) Evaluation of LSI in word-based and syllable-based vector spaces: The results of applying LSI to word-based and syllable-based VSM are shown Fig. 3. (Left: word-based, Right: syllable-based). The left figure shows that LSI did not improve retrieval performance in the word-based vector space. We believe this was caused by the following reason. Because the length of a lecture was about 15 minutes, there were a lot of utterances which were transcribed into sentences by a speech recognizer. These sentences were mapped as one vector using only 14,716 index terms. Therefore, LSI was not able to capture the latent semantic information, such as co-occurrence information and so on, in the word-based vector space because the document vectors are not so sparse. On the other hand, we can see from the right side of Fig. 3 that LSI improved retrieval performance from 0.287 to 0.308 in syllable-based VSM by reducing the dimensions of the document vector. The number of index terms in the syllable-based vector space was 169,363. Because the document vector constructed using these index terms was high dimensional and sparse, there was possibility containing the noise in the document. Therefore, dimension reduction using LSI reduced the influence of these noises and improved retrieval performance. In the future, we plan to investigate the latent semantic information, extracted by this method and make sure the effectiveness of the method.

3) Evaluation of distance combinations: This section describes experimental results regarding our distance combination method. We combined the distances of calculated in binary-weighted of word-based VSM, the TF-IDF-weighted word-based VSM with query expansion ($\alpha = 0.9$), and the TF-IDF-weighted syllable-based VSM. The combined distances were calculated using the following equation:

$$\begin{aligned} S &= (1-\beta_1-\beta_2)s_b(\boldsymbol{d}_i,\boldsymbol{q}) \\ &\quad +\beta_1 s_e(\boldsymbol{d}_i,\boldsymbol{q})+\beta_2 s_s(\boldsymbol{d}_i,\boldsymbol{q}), \end{aligned} \tag{6}$$

where $s_b()$, $s_e()$ and $s_s()$ indicate the cosine similarity calculated in binary-weighted word-based VSM, the TF-IDF-weighted word-based VSM with query expansion, and the TF-IDF-weighted syllable-based VSM, respectively. From the results of Section III-B2, we discovered that LSI improved retrieval performance in the syllable-based document space. Hence, we used two types of distances calculated from the syllable-based VSM, with and without LSI. Table I shows the experimental results of the proposed distance combination method. The best MAP scores were described among various β_1 and β_2 along with the values of β_1 and β_2 in Table I. We also show the results for word-based TF-IDF with query expansion for comparison, which is corresponding to the case with $(\beta_1, \beta_2) = (1.0, 0.0)$. Comparing the result for query expansion method in Table I with the distance combination methods, we can

TABLE I
RETRIEVAL RESULTS FOR DISTANCE COMBINATION

method	(β_1, β_2)	MAP
Word-based VSM with query expansion	(1.0,0.0)	0.343
Distance combination without LSI	(0.3,0.2)	0.389
Distance combination with LSI	(0.2,0.1)	0.394

TABLE II
RESULTS OF FORMAL RUN

method	MAP
TF-IDF-weighted word-based VSM	0.398
Word-based VSM with query expansion	0.440
Distance combination method without LSI	0.453
Distance combination method with LSI	0.421

see that the distance combination methods improved MAP scores. Therefore, we can conclude that distance combination methods are useful for spoken document retrieval. Moreover, applying LSI to syllable-based VSM also improves the retrieval performance of the distance combination method.

C. Results of Formal Run

In the previous section, we investigated the optimal parameters using the development set. In this section, we conduct spoken document retrieval experiments using the "Formal Run" open test set from NTCIR-9.

1) Experimental Conditions: In this experiment, we used 86 queries which were not same as the queries used with the development set. Other experimental conditions were the same as those used with the development set. We compared the following four methods which showed high spoken document retrieval performances in previous experiments:

- TF-IDF-weighted word-based VSM
  With this method, the number of speech recognition candidates for constructing the vector space was set at 1 and the term weight was TF-IDF.
- Word-based VSM with query expansion
  With this method, the number of speech recognition candidates for constructing the vector space was set to 3 and the term weight was TF-IDF. In addition, the weight parameter for query expansion and the number of web pages for query expansion were set to 0.9 and 21, respectively.
- Distance combination method without LSI
  For the distance combination method, we used the TF-IDF-weighted word-based VSM with query expansion, the binary-weighted word-based VSM, and the TF-IDF-weighted syllable-based VSM. We set the weight parameters of distance combination (β_1 and β_2) to 0.2 and 0.3. Other parameters were the same as for "Word-based VSM with query expansion."
- Distance combination method with LSI
  With this method, we applied LSI to the TF-IDF-weighted syllable-based VSM. The dimension of the vector space was 768 after LSI was applied. We set the weight parameters of distance combination (β_1 and β_2) to 0.2 and 0.1. Other parameters were the same as for "Distance combination method without LSI."

2) Experimental Results: Table II shows the experimental results of Formal Run. Comparing "TF-IDF-weighted word-based VSM" with "Word-based VSM with query expansion" and "Word-based VSM with query expansion" with "Distance combination method without LSI" in Table II, we see the same tendency as when using the development set. Hence, we conclude that query expansion and distance combination are useful methods for spoken document retrieval.

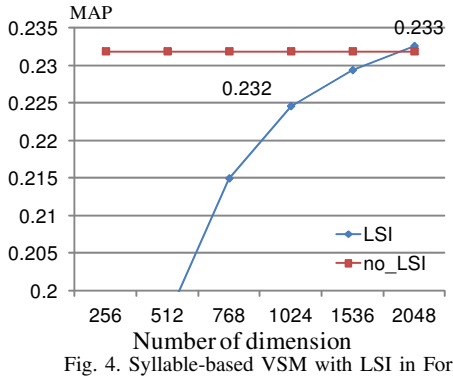

Fig. 4. Syllable-based VSM with LSI in Formal Run

On the other hand, comparing "Distance combination method with LSI" with "Distance combination method without LSI," we can see that LSI does not improve retrieval performance when using the open test set. Moreover, we can see from Table II that "Distance combination method with LSI" degrades retrieval performance of "Word-based VSM with query expansion," although this method gave the highest performance with the development set. To examine these results, we investigated the relationship between retrieval performance and the number of dimensions in the reduced vector spaces. The results of this investigation are shown in Fig. 4. According to these results, retrieval performance was degraded by reducing the number of dimensions of the open test set, although dimension reduction by applying LSI increased the MAP score with the development set. In this experiment, by applying LSI ,the number of dimensions of the syllable-based vector space was reduced from 169,363 to 768, which was the optimal value for the development set. However, the performance of syllable-based VSM was degraded. From these results we can see that we need to do further investigation of suitable LSI parameters, as well as of the parameters for the distance combination method.

IV. CONCLUSION

In this paper, we proposed a spoken document retrieval method using vector space models in multiple document spaces. First, using the proposed method we constructed multiple document vector spaces, one of which was based on continuous word speech recognition results and the other on continuous syllable speech recognition results. Then query expansion was applied to the word-based document space. In addition, we proposed applying latent semantic indexing (LSI) to the syllable based space to reduce dimensionality of the spaces, using implicitly defined semantics. Finally, the proposed method combined the distances between the query and target spoken documents in various spaces to rank the documents.

To evaluate our proposed method, we conducted spoken document retrieval experiments using the NTCIR-9 SpokenDoc data set. The results showed that query expansion and combination of the distances improved retrieval performance when using both the development set and the open test set. However, using LSI in the syllable-based document space did not improve the retrieval performance on the open test set, although this did improve retrieval performance when using the development set.

In future work, we plan to study an optimal parameter decision method for performing distance combination calculations.

ACKNOWLEDGMENT

This work has been partially supported by the 'Core Research for Evolutional Science and Technology' (CREST) project of Japan Science and Technology Agency (JST).

REFERENCES

[1] Turnen, V.T., "Reducing the Effect of OOV Query Words by Using Morph-Based Spoken Document Retrieval", *Proc. of Interspeech*, pp.2158–2161 2008.

[2] Iwata, K., Shinoda, K. and Furui, S., "Robust Spoken Term Detection Using Combination of Phone-Based and Word-Based Recognition", *Proc. of Interspeech*, pp.2195–2198 2008.

[3] Saraclar, M. and Sproat, R., "Lattice–Based Search for Spoken Utterance Retrieval", *HLT–NAACL*, pp.129–126 2004.

[4] Deerwester, S., Dumais S.T., Furnas, G.W., Landauer, T.K., "Indexing by Latent Semantic Analysis", *Journal of American Society for Information Science*, pp.391–407 1990.

[5] Wiemer–Hastings, P., "Latent Semantic Analysis", *In Proceedings of the Sixteenth International Joint Congress on Artifical Intelligence*, pp.932–937 2004.

[6] Rocchio, J., "Relevance Feedback in Information Retrieval", *Salton G. (Ed.), The SMART Retrieval System. Englewood Cliffs, N.J: Prentice Hall*, pp.313–323 1971.

[7] Akiba, T., Nishizaki, H., Akikawa, K., Kawahara, T. and Matsui, T., "Overview of the IR for Spoken Document Task in NTCIR–9 Workshop", *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Question Answering and Cross-lingual Information Access*, pp.223–235 2011.

[8] Maekawa, K., Koiso, H., Furui, S. and Isahara, H., " Spontaneous speech corpus of Japanese ", *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC)*, pp.947–952 2000.

[9] "JSON/Atom Custom Search API", Online: http://code.google.com/apis/customsearch/v1/overview.html.