Spoken Document Retrieval Experiments for SpokenDoc at Ryukoku University (RYSDT)

Hiroaki NANJO
Faculty of Science and Technology, Ryukoku University, Japan
nanjo@rins.ryukoku.ac.jp

Kazuyuki NORITAKE
Graduate School of Science and Technology, Ryukoku University, Japan
noritake@nlp.i.ryukoku.ac.jp

Takehiko YOSHIMI
Faculty of Science and Technology, Ryukoku University, Japan
yoshimi@rins.ryukoku.ac.jp

ABSTRACT

In this paper, we describe spoken document retrieval systems in Ryukoku University, which were participated in NTCIR-9 IR for Spoken Documents ("SpokenDoc") task. In NTCIR-9 "SpokenDoc" task, there are two subtasks: "Spoken term detection (STD) subtask" and "Spoken document retrieval (SDR) subtask". We participated in the both subtasks as team RYSDT. In this paper, first, our STD systems are described, and then, our SDR systems are described.

Categories and Subject Descriptors

H3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords

NTCIR-9, Spoken term detection, Spoken document retrieval

Team Name: [RYSDT]
Subtask/Languages: [Spoken Term Detection] [Spoken Document Retrieval] / [Japanese]
External Resources Used: N/A

1. INTRODUCTION

With the advance of high-speed networks and mass storage, a great deal of audio contents, such as podcasts, TV news, home videos, and lecture/presentation videos, can be easily stored and published. In some universities, lecture videos are actually published in the open domain through web pages. Since such audio contents continue to increase, we need robust methods that can deal with them. Spoken document retrieval (SDR) and spoken term detection (STD), which process such huge amounts of spoken data for efficient search and browsing, are promising.

We have studied both STD [1] and SDR [2][3]. In NTCIR-9, Spoken Documents ("SpokenDoc") task is defined, which covers both STD and SDR. In this paper, our STD and SDR approaches and the results of their applications to NTCIR-9 task are described. Our team name is RYSDT in NTCIR-9 SpokenDoc [4].

2. SPOKEN TERM DETECTION SYSTEMS

Spoken term detection (STD) is the process finding the positions of a query term from the set of spoken documents.

Based on the definition of Kaneko et al. [5], we regard STD as a line detection problem in a term-ASR result image file in which each pixel holds the distance between the syllables in the query term and the ASR results. If there are no ASR errors, a line appears in such an image, and thus we can detect the term positions by detecting lines in it. However, such images essentially include ASR errors, especially for out-of-vocabulary (OOV) word segments. Based on ASR errors, in an image, some line components are off their positions or have even disappeared. For these reasons, conventional line detection-based STD/keyword spotting [5] cannot achieve sufficient accuracy. For a robust line detection-based STD, line detection methods from noisy images must be investigated.

Based on these backgrounds, we proposed image processing filters for line detection-based STDs [1]. Specifically, we proposed a noise removal filter and a line enhanced filter to convert a noisy image in which the target line is difficult to find to a line-detectable image in which the target line is easy to find.

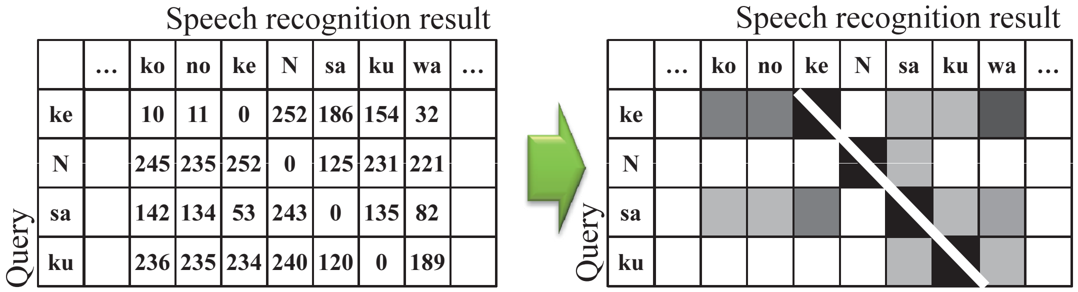
For the NTCIR-9 STD subtask in SpokenDoc, we applied above described line detection-based STD systems which use two image filters. In this section, we described the details of our systems and their STD results.

2.1 Overview of Line Detection-based STD and its Problems

First, we describe an overview of STD based on line detection in image files. One problem with STD is finding speech segments that contain a query term. Indexing by ASR and flexible pattern matching between a query and the index are necessary. As for flexible pattern matching, line detection in image files has been proposed [5]. For a given query (p-syllables) and the ASR result (q-syllables), a $q \times p$ grayscale digital image file is generated. Each element (pixel) $E_{i,j}$ holds a normalized distance (0 to 255) between the i-th syllable of the query and the j-th syllable of the ASR result, which reflects how confusing the syllables are. If there are no ASR errors, a 45-degree line appears in such a grayscale image, and therefore, we can detect the terms by detecting such lines in the image. Figure 1 shows an example of line detection-based STD in query-ASR result grayscale digital images.

Hough transform is one of the most well-known methods for automatic detection of lines in images. Since only a 45-degree line is a target line for STD tasks, line detection-based STD is defined as a process that finds every j such that averaged cumulative distance M_j is smaller than a given threshold α:

$$M_j < \alpha \quad (1 \le j \le q - p) \rightarrow \text{a term exists}, \qquad (1)$$



	…	ko	no	ke	N	sa	ku	wa	…
ke		10	11	0	252	186	154	32	
N		245	235	252	0	125	231	221	
sa		142	134	53	243	0	135	82	
ku		236	235	234	240	120	0	189	

Figure 1: Overview of line detection-based STD

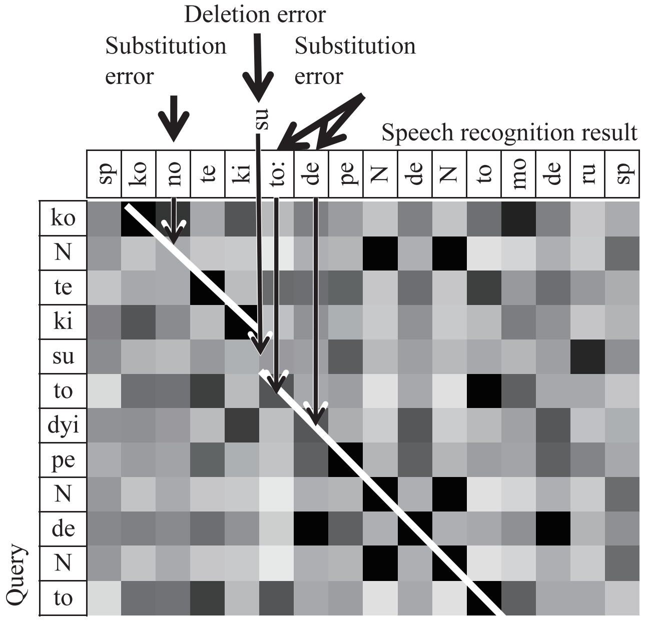

Deletion error
Substitution error
Substitution error
su
Speech recognition result
sp ko no te ki to: de pe N de N to mo de ru sp
ko N te ki su to dyi pe N de N to
Query

Figure 2: Problems for line detection caused by ASR error

where

$$M_j = \frac{1}{p}\sum_{i=1}^{p} D_{i,i+j-1}.$$

Here, p and q are the lengths of the query and the ASR results, that correspond to the image height and width, respectively. $D_{i,j}$ shows a value of (i,j)-pixel that reflects the distance between the i-th syllable of the query and the j-th syllable of the ASR result.

Next, we discuss the main problem of the above line detection-based STD. When the target spoken terms are mis-recognized in a frontend ASR system, line detection errors might be caused. Figure 2 shows a query-ASR result grayscale image when ASR errors are contained in a target spoken term. The ASR errors are classified into three types: substitution error, deletion error, and insertion error. For substitution error, line detection-based STD works robustly since some pixel values are slightly overestimated and gaps appear in a target line that does not generally affect the Hough transformation. On the contrary, insertion and deletion errors cause a significant problem for line detection-based STD. When insertion/deletion errors occur, the target line shifts to the right/down-side from the error point. Therefore, simple line detection based on formulation (1) fails.

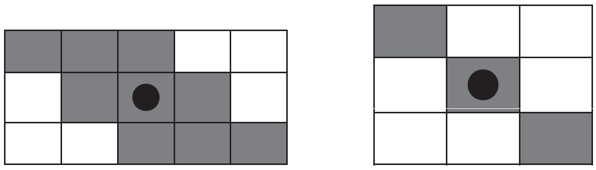


Figure 3: Proposed line enhanced filter (left) and noise removal filter (right)

2.2 Image Processing Filters for Speech Recognition Errors

2.2.1 Line Enhanced Filter

Here, we describe a line enhanced filter that is used for recovering the shifts and gaps of lines caused by ASR errors. We propose a filter that can cope with deletion and substitution errors. The filter, which is shown on the left side of Figure 3, replaces the value of the center pixel (black circle) with the mean of the three highest pixel values in the gray area. This is based on the assumption that even if deletion or insertion errors occur, at least three pixels composing a target line are included in the gray area. An example of line enhancement by the filter is shown in the upper part of Figure 4. We can see a bold line for the line part. With the line enhanced filter, lines are easily detectable even if deletion or insertion error exists. However, it causes negative effects. By adopting this filter, pseudo lines might be detected since some image noises (pixels that have quite different values from their surroundings) are enhanced (example: upper right of Figure 4). Therefore, to solve this problem, we also introduced a noise removal filter that is described in the following section.

2.2.2 Noise Removal Filter

Next, we describe a noise removal filter. Even though smoothing and median filters are common, they remove lines that we want to detect. Thus, we study a filter that removes only the noises and maintains target lines. We propose a kind of median filter that replaces the value of a center pixel with the median of the pixel values of the gray part (right side of Figure 3). In Figure 4 in the lower part, the noise removal result with this filter is shown. Noise pixels (black pixels in upper right) have been thinned. We confirmed that only the pixels around the target line are well enhanced

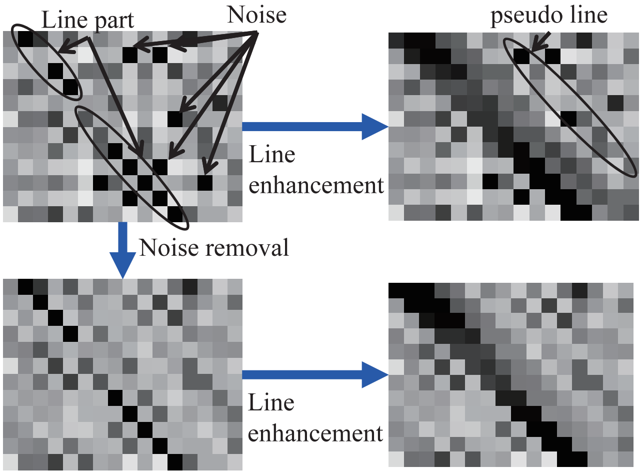


Figure 4: Example of line enhanced and noise removal filtering

compared to the result without a noise removal filter before the line enhanced filter (see upper/lower parts of Figure 4).

2.3 Spoken Term Detection Algorithm

First, we describe our STD algorithm. Figure 5 shows a block diagram. A query term that consists of p-syllables is input, and then for each ASR result of k-th utterance U_k, a query-utterance syllable-distance map (image file) is generated. Next, line detection is performed to retrieve a term. A term is included in U_k where j exists such that $1 \leq j \leq q - p$ and $M_j < \alpha_1$ (formula (1)). If no lines are detected, another line detection is performed for a image enhanced by the proposed filters, in which another threshold α_2 is used. The STD process is performed for all utterances (U_k: $k = 1 \ldots N$), and a list of those that include the term is output.

Then, we defined syllable-distance that gives each $D_{i,j}$ ((i, j)-pixel value). For it, we used the mean of the normalized Bhattacharyya distances (NBD) of phoneme HMMs that consist of syllables:

$$\text{NBD} = 255\ (1 - \exp(\beta \cdot \text{Bhattacharyya-distance})). \quad (2)$$

Here, normalized parameter β is set to 0.75 empirically.

We scrutinized the detection errors and found different error tendencies based on the length of a search term. For shorter terms, since false detections frequently occur, smaller threshold α (formula (1)) is required. Moreover, filters are not worked well. On the contrary, for longer terms, filters are worked well, and mitigating detection threshold α is effective.

2.4 From-CORE detection task

2.4.1 Submitted STD Systems

ASR results with word 3-gram language model, which are given by the task organizer, namely, “REF-WORD”, are used. According to the dryrun results for OOV task, we selected the detection methods and the thresholds α_1 and α_2 for each term length. Actually, we constructed three systems as follows.

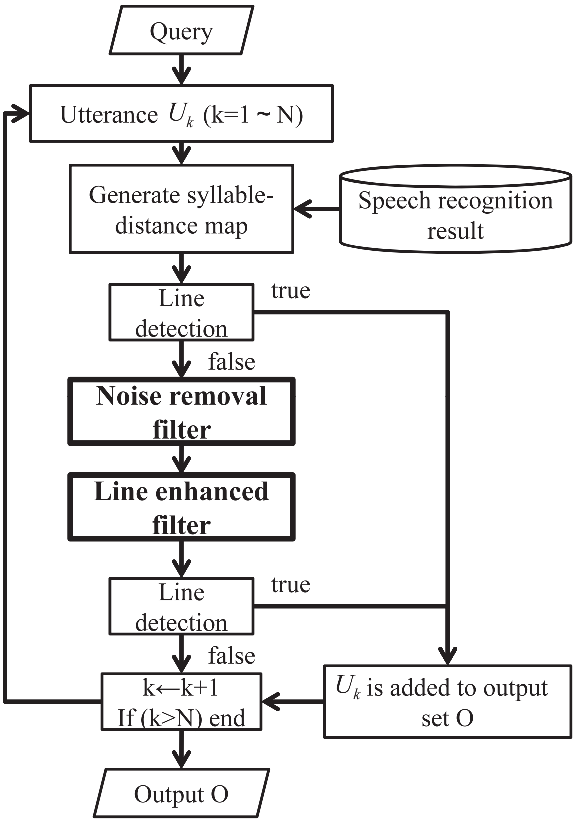


Figure 5: Block diagram of STD algorithm

- sys1 (RYSDT-1):
  - terms ($\leq$ 5 syls): w/ filters, $\alpha_1 = 50$, $\alpha_2 = 30$
  - terms (6, 7 syls): w/ filters, $\alpha_1 = 50$, $\alpha_2 = 50$
  - terms ($\geq$ 8 syls): w/ filters, $\alpha_1 = 50$, $\alpha_2 = 70$
- sys2 (RYSDT-2):
  - terms ($\leq$ 5 syls): w/o filters, $\alpha_1 = 30$, $\alpha_2 = N/A$
  - terms (6, 7 syls): w/ filters, $\alpha_1 = 50$, $\alpha_2 = 50$
  - terms ($\geq$ 8 syls): w/ filters, $\alpha_1 = 70$, $\alpha_2 = 70$
- sys3 (RYSDT-3):
  - terms ($\leq$ 5 syls): w/o filters, $\alpha_1 = 50$, $\alpha_2 = N/A$
  - terms (6, 7 syls): w/ filters, $\alpha_1 = 50$, $\alpha_2 = 50$
  - terms ($\geq$ 8 syls): w/ filters, $\alpha_1 = 50$, $\alpha_2 = 50$

In system 1, we performed STD before image filters using $\alpha_1 = 50$, and after image filters, we performed STD with term length dependent thresholds $\alpha_2 = 30$, 50 and 70 for terms which consist of less than 6 syllables, 6 or 7 syllables, and more than 7 syllables, respectively. The result is our intended result (marked ”YES”). Submitted results included utterances marked ”NO”, which are detected with larger α_2 value.

In system 2, for terms which consist of less than 6 syllables, using $\alpha_1 = 30$, we just perform STD without image filters. For terms which consist of 6 or 7 syllables and more than 7 syllables, STD is performed using $\alpha_1 = \alpha_2 = 50$ and

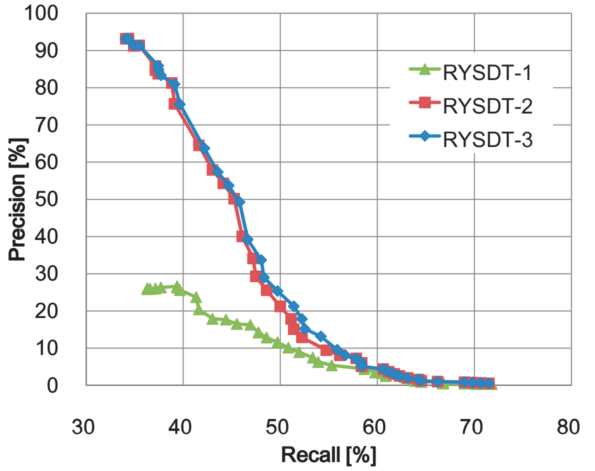


Figure 6: STD Result for CORE task (RYSDT)

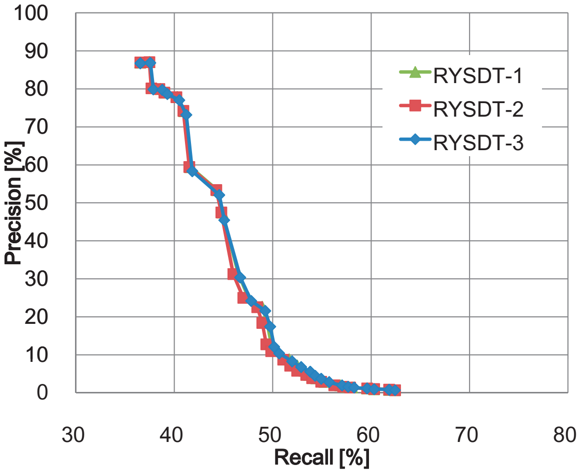


Figure 7: STD Result for ALL task (RYSDT)

Table 1: STD Result for CORE task (RYSDT)

	Sys-1	Sys-2	Sys-3
F-measure (micro average)	31.8	52.6	52.1
MAP (macro average)	0.393	0.469	0.468
Retrieval time	3m26s	2m26s	2m25s

Table 2: STD Result for ALL task (RYSDT)

	Sys-1	Sys-2	Sys-3
F-measure (micro average)	53.1	53.0	53.1
MAP (macro average)	0.431	0.426	0.434
Retrieval time	22m51s	22m37s	23m08s

$\alpha_1 = \alpha_2 = 70$, respectively. The result is our intended result (marked ”YES”). Here, submitted results also included utterances marked ”NO”, which are detected with larger α_2 value.

System 3 is almost the same with system-2, just different in using unique threshold ($\alpha_1 = \alpha_2 = 50$). Here also, the result is our intended result (marked ”YES”), and submitted results included utterances marked ”NO”, which are detected with larger α_2 value.

These α values are normalized to the STD score among 0 to 1 according to the following equation.

$$\text{STD score} = \frac{255 - \alpha}{255} \tag{3}$$

2.4.2 Results

The results (Recall-Precision curve) for from-CORE detection task are shown in Figure 6 and Table 1. The best F-measure is 31.8, 52.6, and 52.1 for system-1 to -3, respectively. System-2 and -3 are almost the same and both of them outperform system-1. The MAPs are 0.393, 0.468, and 0.469, respectively. It shows that for shorter terms, proposed image filters had a bad effect

We did not achieve higher STD performances although the recall rate is more than 70% and is not insufficient. One possible reason is that the STD ranking score by our system is not so reliable. In our system, STD is performed two times before and after image filters. Although STD score after image filters is higher than before image filters for almost all utterances, utterances once detected before image filters were removed from the utterance set to be detected in second STD and they had lower STD ranking scores.

Retrieval time is 2m25s to 3m26s for 50 term detection in total using Xeon 3.20GHz with 4GB memory machine. Our systems required about 2.9 to 4.1 seconds for each query.

2.5 From-ALL detection task

2.5.1 Submitted STD Systems

Here, ASR results with word 3-gram language model, which are given by the task organizer, namely, “REF-WORD”, are also used. We constructed three systems as follows. As for these systems, only system-1 is different from systems for CORE. Specifically, the difference is NOT application of image filters for shorter terms in system-1.

- system-1 (RYSDT-1):
  - terms ($\leq$ 5 syls): <u>w/o filters, $\alpha_1 = 50$, $\alpha_2 = N/A$</u>
  - terms (6, 7 syls): w/ filters, $\alpha_1 = 50$, $\alpha_2 = 50$
  - terms ($\geq$ 8 syls): w/ filters, $\alpha_1 = 50$, $\alpha_2 = 70$
- system-2 (RYSDT-2):
  Same with system-2 for CORE detection task
- system-3 (RYSDT-3):
  Same with system-2 for CORE detection task

2.5.2 Results

The results (Recall-Precision curve) for from-ALL detection task are shown in Figure 7 and Table 2. The best F-measure is 53.1, 53.0, and 53.1 for system-1 to -3, respectively. The MAPs are 0.431, 0.426, and 0.434, respectively. All systems are almost the same performance. It shows that it is better to perform STD without proposed image filters for shorter terms.

The performances of system-2 and -3 are almost close to CORE-detection results. It shows that our systems have consistency and robustness for data size.

Retrieval time is 22m51s to 23m08s for 50 term detection in total using Xeon 3.20GHz with 4GB memory machine.

Our systems required about 27 seconds for each query. The size for ALL set is about 10 times of CORE set size, and our systems require much retrieval time according to database size linearly.

3. SPOKEN DOCUMENT RETRIEVAL SYSTEMS

Spoken document retrieval (SDR) is a process finding the spoken document itself or short portions (passages) of spoken document which are relevant to the query. For the SDR task in NTCIR-9, search target are Japanese oral presentations in academic conferences and simulated presentations. Since each presentation has a longer duration about 12 minutes to 1 hour, searching each presentation is not preferable since we cannot access a specific scene which we want to know even if the suitable presentations are perfectly retrieved. Therefore, NTCIR-9 SDR defined not only lecture unit retrieval task, but also passage unit retrieval task. We tried to search both lecture unit and passage unit based on an orthodox vector space model (VSM).

3.1 Problem in vector space modeling in Japanese SDR

For a VSM-based SDR, appropriate indexing is significant. Automatic speech recognition (ASR) is performed to make index terms, which essentially contain ASR errors. Therefore, studies of indexing terms that are robust to ASR errors are necessary. In Japanese text, no space is put between words, and word units are ambiguous. Thus, studies of indexing units are also important. Based on this background, we have investigated several indexing units in Japanese SDR [2] including morpheme unit, character n-gram unit, and phone n-gram unit. We have found that morphemes is suitable for indexing unit and only nouns baseforms of verbs are suitable for index terms.

For the NTCIR-9 SDR subtask in SpokenDoc, we applied above described VSM based SDR systems. In this section, we described the details of our systems and their SDR results.

3.2 SDR system based on vector space model

3.2.1 Indexing unit and terms

In our system, Japanese morpheme is defined as an indexing unit. As for indexing terms, baseforms of verbs are suitable for index terms. To extract such indexing terms from the automatically transcribed lecture texts (ASR results), we performed morphological analysis with Japanese morphological analysis system ChaSen Ver2.2.1 with ipadic-2.4.1. Some character strings are not defined by the ChaSen and are regarded as unknown words, which are regarded as noun in this work. Then, only nouns and verbs (baseform) recognized by the ChaSen are used for index terms.

3.2.2 Retrieving algorithm

We constructed SDR system based on a VSM and indexing units based on nouns and verbs. In VSM, queries and documents to be retrieved are represented by vectors whose elements correspond to each index term frequencies in each query/document, and vector distance is used as a query-document similarity score. According to the similarity scores, SDR systems output documents. In this work, as a similarity measure between vectors, SMART[6] is used and the Generic Engine for Transposable Association (GETA)[7] is used for constructing VSM-based SDR systems.

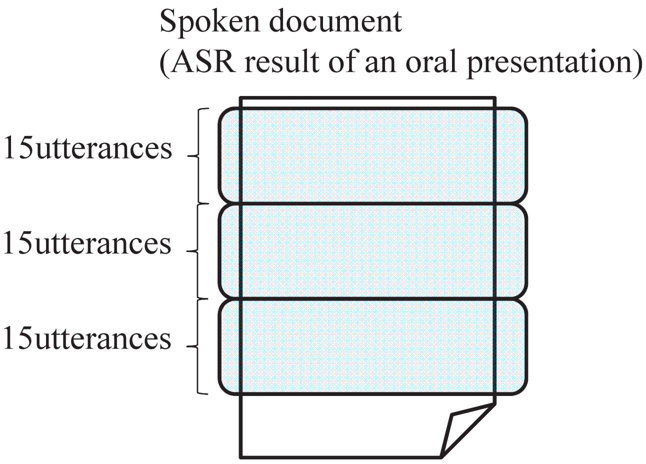


Figure 8: 15-utterances-based unit

Our system differs only at used transcription “OWN” and selection of index terms (stop words) from baseline system[4].

3.3 Submitted SDR system

3.3.1 Lecture retrieval system

First, the results for lecture retrieval task are described. In a lecture retrieval task, search target is an each lecture (total 2702). Each lecture transcription (ASR results) is regarded as a document and VSM-based SDR system is constructed.

3.3.2 Passage retrieval system

Next, the results for passage retrieval task are described. In a passage retrieval task, search target is a short part (utterance sequences of arbitrary length) in lectures. Although participants are requested to detect start and end points of such parts, determining such arbitrary length passages is time consuming. Here, we used a uniformly automatically segmented unit as a passage unit instead of such arbitrary length unit. Actually, as shown in Figure 8, we just divided oral presentation speech from its beginning into several segments which consist of 15 sequential utterances, which is introduced in the Japanese SDR test collection [8]. We regarded each 15 sequential utterance as a passage/document (total 60202), and VSM-based SDR system is constructed.

Our system differs only at used transcription “OWN” and selection of index terms (stop words) from baseline system[4].

3.4 SDR results

3.4.1 Lecture retrieval

For indexing, we used our own ASR result “OWN” which is identical to [8]. The WER is about 5% to 40% for each lecture and 20% in average. For each query, we tried to retrieve 1000 documents.

The results are listed in Table 3. Mean average precision (MAP) is 0.539. Standard deviation is 0.246. Histogram of averaged precisions for each query is shown in Figure9. An

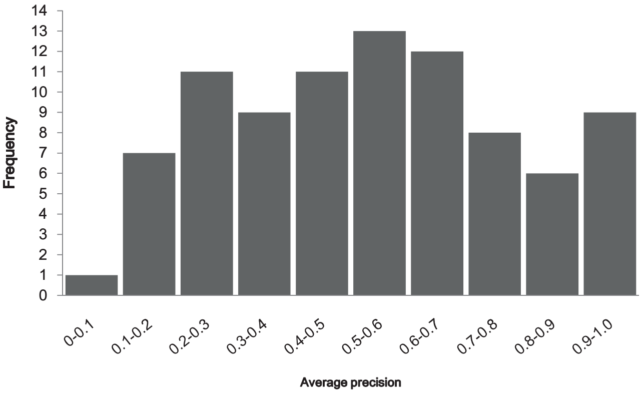


Figure 9: Histogram of SDR performance (averaged precision) for lecture retrieval task

Table 3: Lecture retrieval result (RYSDT)

	average	standard deviation
averaged precison	0.539 (MAP)	0.246
recall	0.951	0.089

Table 4: Passage retrieval result (RYSDT)

uMAP	0.0751
pwMAP	0.0725
fMAP	0.0650

average precision seems to follow uniform distribution.

3.4.2 Passage retrieval

Here, we also used our own ASR result "OWN" which is identical to [8]. Above described 15-utterance-based unit is regarded as a document, and for each query, we tried to retrieve 1000 documents.

The results are listed in Table 4. uMAP, pwMAP, and fMAP are 0.0751, 0.0725, and 0.0650, respectively. Our system always outputs 1000 of 15-sequential-utterances (total 15000 utterances), therefore, we did not achieve higher uMAP and fMAP. In our system, retrieved document is a uniformly divided segment, and we did not consider that the center of each segment is relevant to the query. Therefore, we did not achieve higher pwMAP.

4. CONCLUSIONS

We participated in NTCIR-9 Spoken Documents ("SpokenDoc") task as a team "RYSDT". In this paper, our STD and SDR systems, which are participated in NTCIR-9 SpokenDoc STD subtask and SDR subtask, were described. As for STD, line-detection based STD system with image filters was described and we showed the effectiveness of the image filters. As for SDR, vector space model (VSM) based SDR system considering the selection of indexing terms was described. Our system showed the sufficient performance for lecture retrieval task. For a passage retrieval task, we just regarded a pre-divided segment as a document and retrieved such a segment. We confirmed that the simple method did not work well, and there is a room for investigation of passage retrieval.

5. REFERENCES

[1] Kazuyuki Noritake, Hiroaki Nanjo, and Takehiko Yoshimi. Image processing filters for line detection-based spoken term detection. In *the 12th Annual Conference of the International Speech Communication Association (INTERSPEECH 2011)*, pages 2125–2128, 2011.

[2] Koji Shigeyasu, Hiroaki Nanjo, and Takehiko Yoshimi. A study of indexing units for japanese spoken document retrieval. In *10th Western Pacific Acoustics Conference (WESPAC X)*, 2009.

[3] Hiroaki Nanjo, Yusuke Iyonaga, and Takehiko Yoshimi. Spoken Document Retrieval for Oral Presentations Integrating Global Do cument Similarities into Local Document Similarities. In *Proc. Interspeech (INTERSPEECH 2010)*, pages 1285–1288, 2010.

[4] Tomoyosi Akiba, Hiromitsu Nishizaki, Kiyoaki Aikawa, Tatsuya Kawahara, and Tomoko Matsui. Overview of the IR for Spoken Documents Task in NTCIR-9 Workshop. In *NTCIR-9*, 2011.

[5] Taisuke Kaneko and Tomoyosi Akiba. Metric Subspace Indexing for Fast Spoken Term Detection. In *Proc. Interspeech*, pages 689–692, 2010.

[6] Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted Document Length Normalization. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 21–29, 1996.

[7] Shingo Nishioka Makoto Iwayama Toru Hisamitsu Osamu Imaichi Akihiko Takano, Yoshiki Niwa and Hirofumi Sakurai. Information Access Based on Associative Calculation. In *SOFSEM 2000: Theory and Practice of Informatics*, Lecture Notes in Computer Science, pages 15–35, 2000.

[8] Tomoyosi Akiba, Kiyoaki Aikawa, Yoshiaki Itoh, Tatsuya Kawahara, Hiroaki Nanjo, Hiromitsu Nishizaki, Norihito Yasuda, Yoichi Yamashita, and Katunobu Itou. Construction of a test collection for spoken document retrieval from lecture audio data. *Journal of Information Processing*, 17:82–94, 2009.