IEICE TRANS. INF. & SYST., VOL.E88–D, NO.3 MARCH 2005

PAPER Special Section on Corpus-Based Speech Technologies

Improving Keyword Recognition of Spoken Queries by Combining Multiple Speech Recognizer's Outputs for Speech-driven WEB Retrieval Task

Masahiko MATSUSHITA[†], *Nonmember*, **Hiromitsu NISHIZAKI**[††a)], *Member*, **Takehito UTSURO**[†††b)], *Nonmember*, *and* **Seiichi NAKAGAWA**[††††c)], *Member*

SUMMARY This paper presents speech-driven Web retrieval models which accept spoken search topics (queries) in the NTCIR-3 Web retrieval task. The major focus of this paper is on improving speech recognition accuracy of spoken queries and then improving retrieval accuracy in speech-driven Web retrieval. We experimentally evaluated the techniques of combining outputs of multiple LVCSR models in recognition of spoken queries. As model combination techniques, we compared the SVM learning technique with conventional voting schemes such as ROVER. In addition, for investigating the effects on the retrieval performance in vocabulary size of the language model, we prepared two kinds of language models: the one's vocabulary size was 20,000, the other's one was 60,000. Then, we evaluated the differences in the recognition rates of the spoken queries and the retrieval performance. We showed that the techniques of multiple LVCSR model combination could achieve improvement both in speech recognition and retrieval accuracies in speech-driven text retrieval. Comparing with the retrieval accuracies when an LM with a 20,000/60,000 vocabulary size is used in an LVCSR system, we found that the larger the vocabulary size is, the better the retrieval accuracy is.

key words: *speech recognition, machine learning, multiple LVCSR models, WEB retrieval*

1. Introduction

Automatic speech recognition, which decodes the human voice to generate transcriptions, has of late become a practical technology. Speech recognition can be used in real world computer-based applications, specifically, those associated with human language. In fact, a number of speech-based methods have been explored in the information retrieval (IR) community. In previous works on spoken document retrieval, type-inputted queries have been mainly used to search speech archives for relevant speech information. In previous works on speech-driven retrieval, on the other hand, spoken queries have been used to retrieve relevant textual (or possibly speech) information. Initiated partially by the TREC-6 spoken document retrieval (SDR) track [1], various methods have been proposed for spoken document retrieval. However, a relatively small number of techniques have been explored for speech-driven text retrieval. Barnett et al. [2] performed comparative experiments related to speech-driven retrieval. Crestani [3] showed that conventional relevance feedback techniques marginally improved the accuracy for speech-driven text retrieval. These two cases focused solely on improving text retrieval methods and did not address problems in improving speech recognition accuracy.

Along with the NTCIR-3 [4] Web retrieval main task, which was organized to promote conventional text-based retrieval, Fujii et al. [5] organized the "speech-driven retrieval" subtask. Unlike those previous approaches, they [5] integrated continuous speech recognition and text retrieval to improve both recognition and retrieval accuracies in speech-driven text retrieval. Their method used target documents to adapt language models (LMs) and to recognize out-of-vocabulary (OOV) words for speech recognition.

To further improve speech recognition accuracy of spoken queries and then improving retrieval accuracy in speech-driven text retrieval, this paper evaluates the techniques of combining outputs of multiple LVCSR models* [6], [7] for recognition of spoken queries of the NTCIR-3 speech-driven Web retrieval task. As model combination techniques, we experimentally compare high-performance machine learning techniques such as Support Vector Machine (SVM) learning [8] and conventional voting schemes such as ROVER (*Recognizer Output Voting Error Reduction*) [9]–[12]. In addition, we use multiple LVCSR models with an LM whose vocabulary size is 60,000, and compare it with our previous experimental results in which we used an LM with a 20,000 vocabulary size [13].

Figure 1 illustrates the overall framework of our speech-driven text retrieval based on multiple LVCSR

Manuscript received July 7, 2004.
Manuscript revised September 16, 2004.

[†]The author is with the Denso Techno Corporation, Nagoya-shi, 450–0002 Japan.
[††]The author is with the Interdisciplinary Graduate School of Medicine and Engineering, University of Yamanashi, Kofu-shi, 400–8511 Japan.
[†††]The author is with the Graduate School of Informatics, Kyoto University, Kyoto-shi, 606–8501 Japan.
[††††]The author is with the Department of Information and Computer Sciences, Toyohashi University of Technology, Toyohashi-shi, 440–8580 Japan.
a) E-mail: hnishi@yamanashi.ac.jp
b) E-mail: utsuro@pine.kuee.kyoto-u.ac.jp
c) E-mail: nakagawa@slp.ics.tut.ac.jp
DOI: 10.1093/ietisy/e88–d.3.472

*In this paper, we define "an LVCSR model" as a set of a decoder, an language model, and an acoustic model which is trained under various conditions such as triphone, syllable, and kinds of feature. Thus, we regard an LVCSR model as one of various LVCSR systems if decoders used among the LVCSR systems are the same, but the type of acoustic model is different from others.

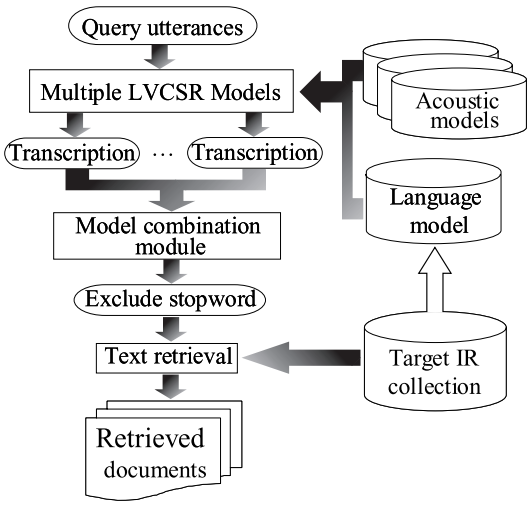


Fig. 1 Speech-driven text retrieval based on multiple LVCSR model combination.

model combination. Query utterances are transcribed individually by each of the multiple LVCSR models, and their outputs are combined by the model combination module. A keyword set for retrieving WEB documents is made by excluding stopwords[†] from the outputs of model combination module. The text retrieval module searches a target IR collection for documents relevant to the query using the keyword set. Beside individual LVCSR models, we evaluated eight models with different decoders and acoustic models, but the same LMs. We used Fujii et al. [5]'s LM and the text retrieval module in the overall framework of Fig. 1, where the LM was trained using the text of the target IR collection.

In this paper, we report the results of our experimental evaluation and show that the techniques of multiple LVCSR model combination can achieve improvement both in speech recognition and retrieval accuracies in speech-driven text retrieval. In addition, comparing with the retrieval accuracies when an LM with a 20,000/60,000 vocabulary size is used in an LVCSR system, we found that the larger the vocabulary size is, the better the retrieval accuracy is.

2. Specification of Japanese LVCSR Models

2.1 Decoders

We use the so-called *Julius* (ver.3.3) decoder among the Japanese LVCSR models. It is provided by the IPA Japanese dictation free software project [14]. We also use the one named *SPOJUS*, which was developed in our laboratory [15], [16]. Both decoders are composed of two decoding passes, where the first pass uses the word bigram, and the second pass uses the word trigram.

2.2 Acoustic Models

The acoustic models among the Japanese LVCSR models are based on a Gaussian mixture HMM. We evaluate phoneme-based HMMs as well as syllable-based HMMs. Speaker-independent acoustic models were trained by using read speech (about 20,000 sentences uttered by 180 male speakers; JNAS).

2.2.1 Acoustic Models with Julius Decoder

In the acoustic models used with the Julius decoder, we evaluate phoneme-based HMMs as well as syllable-based HMMs. The following four types of HMMs are evaluated: i) triphone model, ii) phonetic tied mixture (PTM) triphone model, iii) monophone model, and iv) syllable model. Every HMM is gender-dependent (male). The feature parameters consist of 12 dimensional mel frequency cepstrum coefficients (MFCC), delta 12 dimensions, and delta powers (henceforth "MFCC-frm"). The sampling frequency is 16 kHz, and the frame is shifted by 10 ms at every frame.

A typical triphone HMM consists of 5 states with 3 self-loops and 3 output distributions. Each distribution is composed of 16 Gaussian mixtures having diagonal covariance matrices. The total number of distributions is 2000. On the other hand, a typical syllable-based HMM (124 syllables) consists of 7 states with 5 self-loops and 5 output distributions. Each distribution is composed of 16 Gaussian mixtures having diagonal covariance matrices. The total number of distributions is 600.

2.2.2 Acoustic Models with SPOJUS Decoder

The acoustic models used with the SPOJUS are based on syllable HMMs, which have been developed in our laboratory [17]. The acoustic models are gender-dependent (male) syllable unit HMMs (116 syllables). In our previous works [6], [7], we evaluated the combinations of multiple LVCSR's outputs by SVM and used 18 kinds of acoustic models for the SPOJUS. Considering the experimental results described in [6], [7], we selected 4 types of HMMs which differ in feature parameters and/or self loop transition / duration control [18], [19]. The following feature parameters are used: 24 dimensional mel frequency cepstrum coefficients segmented from 4 successive frames (dimensions reduction by K-L expansion), delta 12 dimensions calculated over 9 successive frames, and delta delta 12 dimensions and delta, delta delta powers (henceforth "MFCC-seg"); 12 dimensional mel frequency cepstrum coefficients, delta, delta delta 12 dimensions, and delta, delta delta powers (MFCC-frm). The sampling frequency is 16 kHz and the frame is shifted by 10 ms at every frame.

Each syllable-based HMM consists of 5 states with 4 self-loops and 4 output distributions. Each distribution is composed of 4 Gaussian mixtures having full covariance matrices.

[†]Stopwords are function words (particle and auxiliary verb), specific clauses which are used in some questions such as "知りたい (I want to know)"and a hiragana character (for example "き", "ひ").

[JP]（サルサ）を（踊れる）ようになる（方法）を知りたい.
[EN]I want to know how to dance ‘‘Sarsa’’.

[JP]（観測）のために（オーロラ）の（発生する）（条件）が知りたい.
[EN]I would like to know the condition of Aurora occuring for its observation.

[JP]（宮部）（みゆき）の（執筆）した（小説）に（対する）（レビュー）が読みたい.
[EN]I want to read the reviews for the novels written by Miyuki Miyabe.

Fig. 2 Examples of the queries (words among parentheses denote keywords).

2.3 Language Model

The LM was prepared by Fujii et al. [5], and trained using the text of the target IR collection. From the 100 GB collection of the target Web text, 20,000/60,000 high-frequency words are independently used to produce a word-based trigram model. The “ChaSen”[†] Japanese morphological analyzer was employed to extract words from the 100 GB Web text collection. To resolve the data sparseness problem, a back-off smoothing method was used, where the Witten-Bell discounting method was chosen for computing back-off coefficients.

3. Evaluation Data Sets

For the NTCIR-3 Web retrieval main task, 105 search topics (queries) were manually produced, for each of which relevance assessment was manually performed with respect to two different document sets, i.e., the 10 GB and the 100 GB collections. In this paper, we used only the 100 GB collection, which includes approximately 10,000,000 documents.

Ten speakers (five adult males/females) were asked to utter the queries of the 105 search topics, which were recorded as spoken queries of the NTCIR-3 speech-driven Web retrieval task. In this paper, we used spoken queries by five male speakers only. The 105 spoken queries were then divided into 52 queries used for training SVM models for model combination, and the remaining 53 queries. Out of the remaining 53 queries, 47 queries (752 words and 329 keywords in total), each of which has reference Web texts within the target 100 GB collection, were used for evaluating both speech recognition and retrieval accuracies. Figure 2 shows examples of the queries used in this paper.

Word correct and accuracy rates of the individual eight LVCSR models, averaged over the five speakers, are summarized in Table 1[††]. Recognition rates are improved by using the LM with the 60,000 vocabulary size comparing with the one with a 20,000 vocabulary size. As shown in Table 2, this is related with OOV rates for test queries. The larger the vocabulary size becomes from 20,000 to 60,000, the lower the OOV rate becomes. Especially, the rate was improved by 9.9% for only keywords included in the queries.

Table 1 Recognition rates for 47 queries in each LVCSR model.

(a) Vocabulary size is 20,000.

LVCSR model		Corr.	Acc.
Julius	monophone	73.2	66.9
	triphone	86.9	78.4
	PTM	85.8	77.5
	syllable	84.3	77.8
SPOJUS	MFCC-seg + duration	85.0	76.4
	MFCC-frm + duration	84.3	76.5
	MFCC-seg + selfloop	81.8	75.0
	MFCC-frm + selfloop	81.8	75.2

(b) Vocabulary size is 60,000.

LVCSR model		Corr.	Acc.
Julius	monophone	76.7	72.6
	triphone	89.4	83.2
	PTM	88.1	82.1
	syllable	87.3	82.9
SPOJUS	MFCC-seg + duration	87.4	82.1
	MFCC-frm + duration	87.5	82.5
	MFCC-seg + selfloop	87.0	82.7
	MFCC-frm + selfloop	86.2	81.5

Table 2 OOV rates in difference of vocabulary size (20,000 and 60,000).

	Voc. size	OOV rate[%]
all words	20,000	4.5
	60,000	1.0
keywords only	20,000	12.7
	60,000	2.8

4. Combining Outputs of Multiple LVCSR Models

4.1 Combination Methods [7]

As techniques for combining outputs of multiple LVCSR models, we experimentally compare SVM learning [8] and conventional voting schemes of ROVER [9]–[12]. The 52 queries are used for training the SVM models[†††]. A Support Vector Machine is trained for choosing the most confident one among several hypothesized words from the outputs of the eight LVCSR models[††††]. As features of the

[†] http://chasen.aist-nara.ac.jp

[††] *correct* = $100 - S - D$ and *accuracy* = $100 - S - D - I$, where S, D, I denote the rate of substitution, deletion and insertion error rates, respectively.

[†††] We evaluate the performance of cross speaker SVM model combination [6], i.e., we perform the speaker-open test in the model combination (cross-validation). An SVM model is trained using 52 queries dictated by four speakers that are not used in the retrieval evaluation. The trained model is evaluated against 47 test queries uttered by the remaining speaker who is not a speaker for the training queries. We perform discretely the retrieval accuracy for 47 test queries uttered by a single speaker. This procedure is repeated for every speaker in turn (exchange a test speaker) and then the obtained results are averaged over the retrieval accuracies from the five speakers.

[††††] We used *TinySVM* (http://chasen.aist-nara.ac.jp/taku/software/TinySVM/) as a tool for SVM learning.

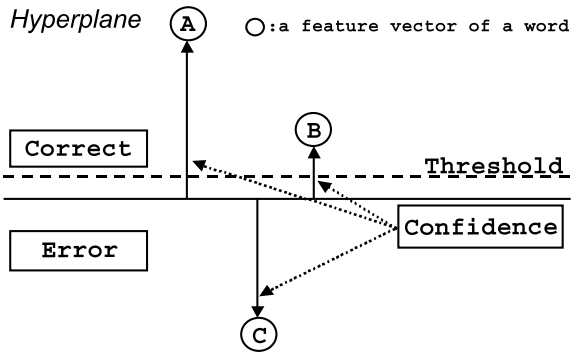


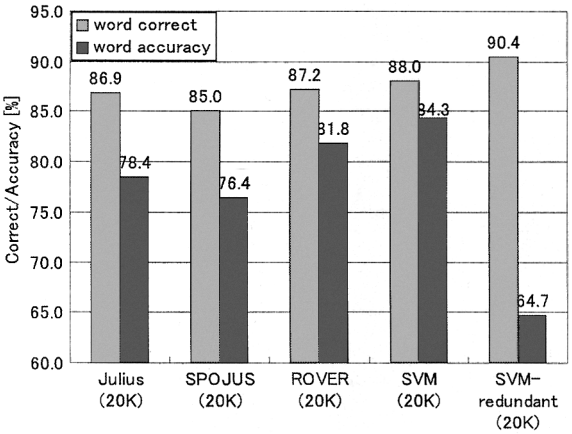
Fig. 3 An outline figure of applying SVM for outputs.

SVM learning, we use the ID of the model which output the word, the part-of-speech of the word, and the word length in syllables of the word[†]. For classes of the SVM learning, we use whether each hypothesized word is correct or incorrect. Since Support Vector Machines are binary classifiers, we regard the distance from the separating hyperplane to each hypothesized word as the word's confidence. The outputs of the eight LVCSR models are aligned by Dynamic Time Warping, and the most confident one among those competing hypothesized words is chosen as the result of model combination. We also require the confidence of hypothesized words to be higher than a certain threshold, and choose those above this threshold as the result of model combination. In Fig. 3, for example, only word "A" is outputted by SVM.

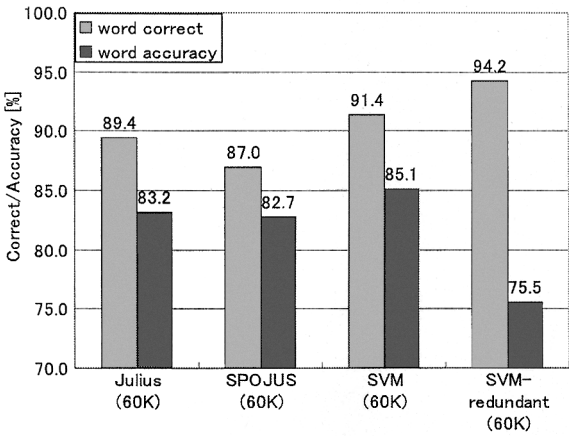
We also evaluate a variant of the above SVM model, namely "SVM-redundant", where its training is exactly the same as the above SVM model, while in the model combination phase, when choosing output words from those competing hypothesized words, SVM-redundant chooses not only the most confident one, but also all the hypothesized words with their confidence values over a certain threshold. SVM-redundant prefers word correct rates to word accuracy rates by simply choosing all those confident hypothesized words competing with each other. In Fig. 3, SVM outputs word "A" and "B" on SVM-redundant.

4.2 Word Recognition Rates of Spoken Queries

Figures 4 and 5 show word correct/accuracy rates as well as keyword correct/accuracy rates of the 47 spoken queries, respectively, where they are averaged over the five speakers. In the experiment using the LM in which vocabulary size is 60,000, we do not perform the ROVER methods as combining outputs from multiple LVCSR models. Because the SVM combination technique outperformed the ROVER in word correct/accuracy. In addition, the effectiveness of the SVM combination has been shown in our previous works [6], [7], [13] in which we compared the SVM with the ROVER on various recognition tasks such as newspaper reading utterances, and news anchor speech. Word correct/accuracy rates in Fig. 4 are those for the whole sentences of the 47 spoken queries, while key-



(a) Vocabulary size is 20,000.



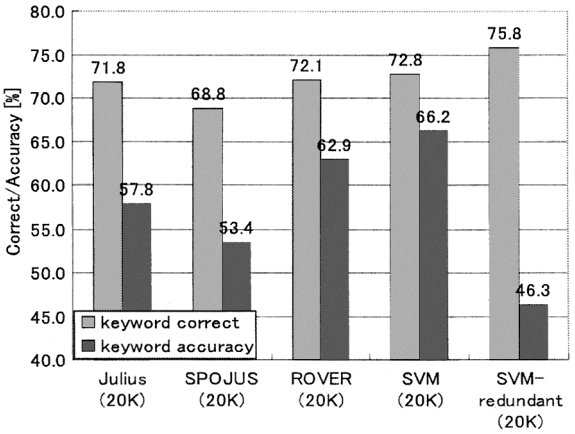
(b) Vocabulary size is 60,000.

Fig. 4 Word recognition rates of spoken queries.

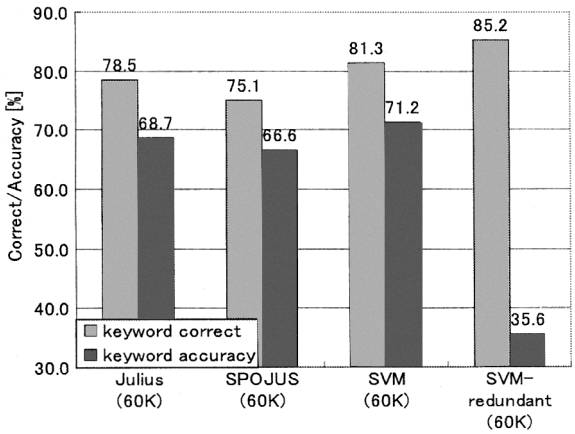
word correct/accuracy rates in Fig. 5 are those after removing stopwords from the speech recognition outputs. Correct/accuracy rates, indicated as "Julius" and "SPOJUS", are the best performing results for each of the two decoders. The numbers in parentheses denote the vocabulary size of the LM used in each LVCSR system.

As clearly seen from these results, the model combination technique such as SVM models serves to improve both word and keyword recognition rates in cases of using both vocabulary sizes. Furthermore, roughly comparing SVM models (i.e., SVM and SVM-redundant) with the conventional voting schemes (i.e., ROVER), SVM models outperform the voting schemes. As described in Sect. 3, the OOV rate for the queries improves with the larger vocabulary size, and consequently, both word and keyword recognition rates improved with the 60,000 LM, compared with the case with the 20,000 LM. Especially, the keyword recognition rates significantly improved in all 5 transcribing methods (Julius, SPOJUS, SVM, SVM-redundant). As we expected, the SVM-redundant improves word/keyword cor-

[†]We also evaluated the effect of acoustic and language scores of each hypothesized word as features of SVM, where their contribution to improving the overall performance was very little.



(a) Vocabulary size is 20,000.



(b) Vocabulary size is 60,000.

Fig. 5 Keyword recognition rates of spoken queries.

rect rates, while damaging its word/keyword accuracy rates.

5. Web Retrieval

5.1 Text Retrieval Model

The text retrieval model was also prepared by Fujii et al. [5]. It is based on an existing probabilistic retrieval method, which computes the relevance score between the translated query and each document in the collection. The similarity $sim(Q, D_i)$ between a query Q and a document D_i is computed as below:

$$sim(Q, D_i) = \sum_t \left(\frac{TF_{t,i}}{\frac{DL_i}{avglen} + TF_{t,i}} \cdot \log \frac{N}{DF_t} \right)$$

Here, t is a keyword in queries. $TF_{t,i}$ denotes the frequency with which the keyword t appears in the document D_i. DF_t denotes the number of documents containing keyword t. N denotes the total number of documents in the collection. DL_i denotes the length of the document D_i (i.e., the number of characters contained in D_i). $avglen$ denotes the average length of documents in the collection.

Given a transcribed keyword sequence, the text retrieval module searches for a target IR collection for relevant documents and sorts them according to the similarities $sim(Q, D_i)$ in descending order. The ChaSen Japanese morphological analyzer was employed to extract words from the 100 GB Web text collection. After excluding stopwords from the word sequence, the remaining words are used as index keywords.

5.2 Evaluation Measures

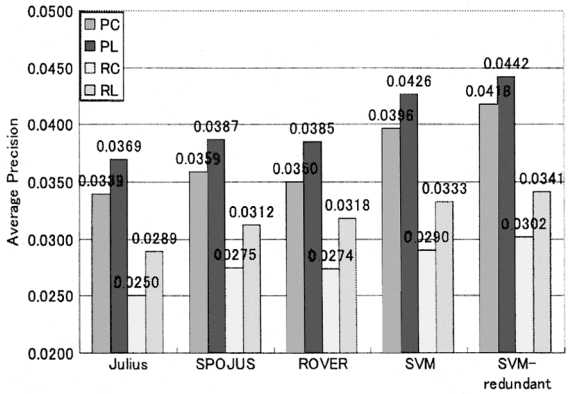
Relevance assessment is performed based on four ranks of relevance, that is, highly relevant, relevant, partially relevant and irrelevant. In addition, unlike conventional retrieval tasks, documents hyperlinked from retrieved documents are optionally used for relevance assessment. To sum up, the following four assessment types are available to calculate average precision values:

- RC : (highly) relevant documents were regarded as correct answers, and hyperlink information was NOT used,
- RL : (highly) relevant documents were regarded as correct answers, and hyperlink information was used,
- PC : partially relevant documents were also regarded as correct answers, and hyperlink information was NOT used,
- PL : partially relevant documents were also regarded as correct answers, and hyperlink information was used.

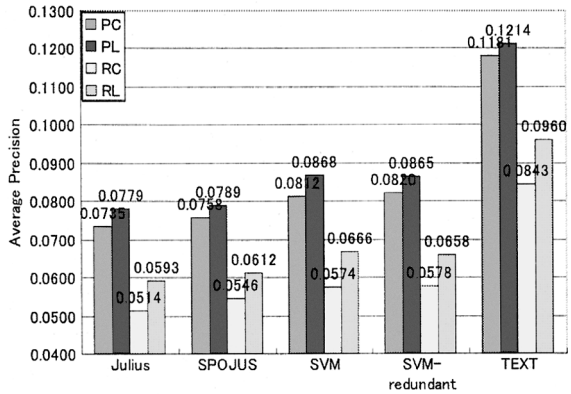
For each of the above four relevance assessment types, we investigate the non-interpolated average precision values [4]. Here, we use the 47 queries to retrieve 1,000 top documents and use the TREC evaluation software to calculate non-interpolated precision values. Finally, those average precision values are further averaged over the five speakers.

5.3 Single LVCSR vs. Multiple LVCSR Models

Evaluation results of Web retrieval experiments are shown in Fig. 6. First, comparing Web retrieval performance between individual LVCSR models and model combination methods in both sizes of vocabulary of the LMs, results for Julius and SPOJUS are the best performing ones for each of the two decoders. Unexpectedly, the best performance for SPOJUS exceeds that of Julius, which is opposite the results of word/keyword recognition rates. This is mainly because word/keyword recognition rates do not depend on the keyword weights computed in the query/document similarity $sim(Q, D_i)$. It could happen that keywords which are correctly recognized by SPOJUS tend to have greater weights than those which are correctly recognized by Julius. The Web retrieval performance of the voting scheme is slightly better than the best performance for Julius, but quite close to that for SPOJUS, while the performance of the SVM models (i.e., SVM and SVM-redundant) is mostly significantly better than those of the individual LVCSR models. Comparing the SVM and the SVM-redundant, the latter outper-

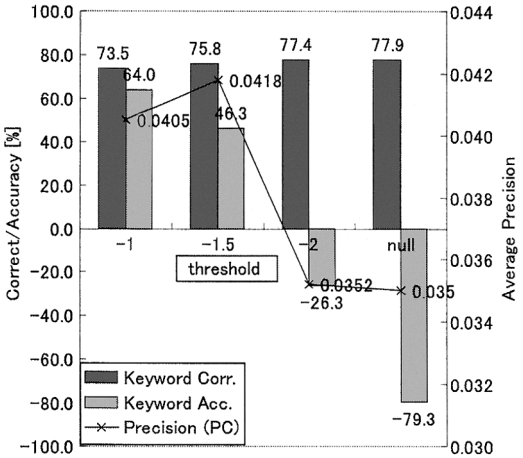


(a) Vocabulary size is 20,000.

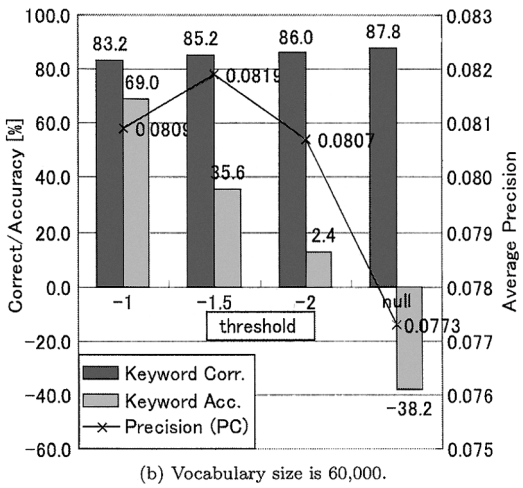


(b) Vocabulary size is 60,000.

Fig. 6 Comparison of Web retrieval performance among model combination methods.



(a) Vocabulary size is 20,000.



(b) Vocabulary size is 60,000.

Fig. 7 Variety of the retrieval performance (PC) when varying the threshold of the SVM-redundant.

forms the former in Fig. 6 (a). However, the performances of the SVM-redundant in case of using the 60,000 LM as shown in Fig. 6 (b) are similar to those of the SVM, because of the large number of keyword insertion errors (49.6%). The threshold is decided experimentaly to get the best retrieval accuracy. The value is independent of a vocabulary size as shown in Fig. 7. With the LM 20,000 vocabulary size, insertion errors are fewer than at 60,000, and its rate is 29.5%. Those results indicate that it is preferable to include as many correctly recognized keywords as possible, even if it damages the keyword accuracy rate. However, too many keyword insertions may further compromise the retrieval accuracies. Interestingly, the improvement of the SVM-redundant over the SVM is greater in "partially relevant" (PC and PL) than in "(highly) relevant" (RC and RL). Since the queries of the SVM-redundant tend to have more keyword recognition errors than the SVM, it seems difficult to improve the performance when (highly) relevant documents must be retrieved. However, this SVM-redundant's weak point, i.e., many insertion errors, may overcome by weighting keywords included in a query. Thus, if only correctly recognized keywords among the large number of keyword candidates are given more weight, the retrieval performance may improve.

Figure 7 shows the retrieval performance (PC) when the threshold for SVM-redundant is varied in some values. The threshold "null" shows the performance by outputting all words located on the correct side of the hyperplane as shown in Fig. 3. The larger the threshold is in negative, the more hypothesized words are outputted, i.e., the keyword recognition accuracy decreases. In both vocabulary sizes, the maximum PC is obtained when the value of the threshold is "−1.5". Those results of PC (−1.5) are identical to the ones in Fig. 6. As shown in Fig. 7, where the larger insertion errors greatly compromise the performance despite decreasing the number of missing keywords, what the unnecessary keywords inserted into the query fatally damages the retrieval accuracies.

5.4 20,000 vs. 60,000 in Vocabulary Size

Next, retrieval performance improved with the 60,000 LM, compared with the case with the 20,000 LM in Fig. 6 (b). This result is easily explained by the differences in the OOV rates as seen in Table 2, where more keywords are covered in the 60,000 LM. In addition, it is also explained by Fig. 8, which shows the retrieval performances when the 22 queries in which OOV keywords are not included are used. In other

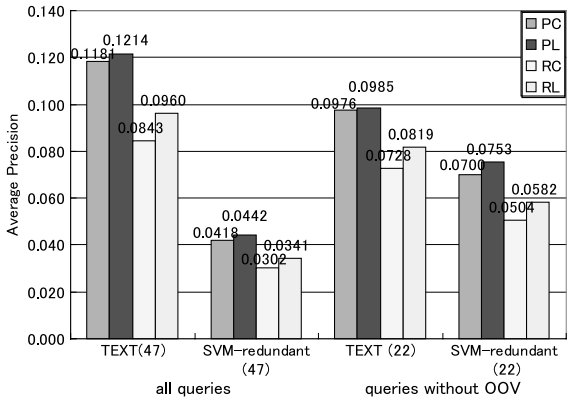


Fig. 8 Effects of OOV words in 20,000 vocabulary size. The numbers in parentheses denote the number of queries.

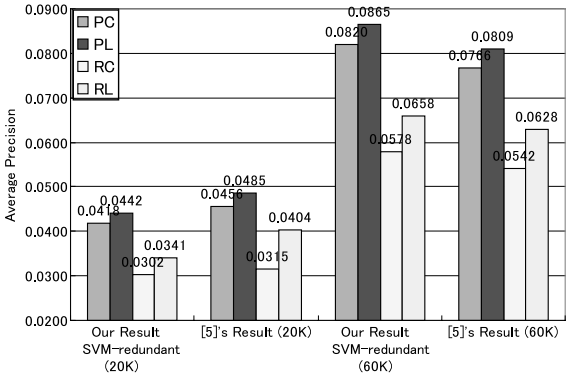


Fig. 9 Comparison of our technique with the previous work.

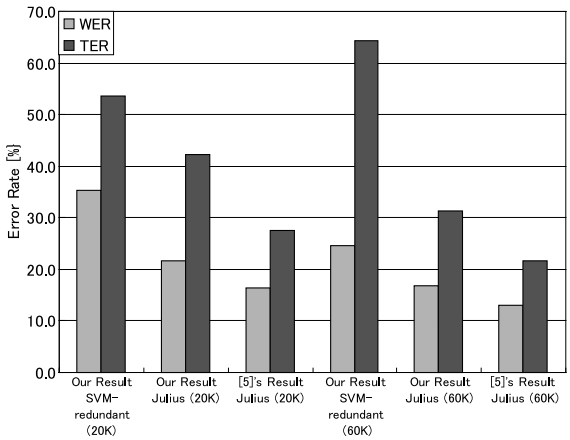


Fig. 10 Comparison of error rates of word and keyword (term).

words, the effect on the retrieval performance depends completely on only the recognition performances of the spoken queries. Although we can not equally compare the 47 queries (including OOV keywords) with the 22 queries (not including OOV keywords), we claim that the OOV keywords adversely affect the retrieval performance rather than degrading recognition performances of the keywords.

It is important for improving the Web retrieval performance to transcribe correctly as many keywords as possible in spoken querie. Whereas comparing with the performances of text queries (those indicated as "TEXT"), it is surprising to see the gaps of their retrieval performance in Fig. 6 (b). The gaps are mainly explained by the difficulty of the Web retrieval task with 100 GB Web text collection. The target of IR 100 GB collection (about 10,000,000 documents) is huge, while the number of documents relevant to a query is very small. Moreover, keywords unrelated to a query may lead to discovery of documents that are irrelevant to the query. Therefore, missing about 15% of the necessary keywords and adding about 50% (= 100 − 35 − 15[%] in Fig. 5 (b)) of unnecessary keywords in a query vastly diminishes retrieval performance.

5.5 Comparison with the Previous Work

Finally, we compare our experimental results with those described in [5], where Fujii et al. conducted an evaluation of the same task. They [5] have experimented on the same task described in this paper. The Julius decoder [14], the 60,000 LM, and the triphone-based acoustic model, as well as the retrieval engine, which are used in [5], are the same as the ones used in this paper. However, our experimental results obtained using the Julius are different from their earlier reported results in spite of using the same decoders and the same retrieval engine. This is because we evaluate the 235 spoken sentences spoken by only five male speakers (47 queries × 5 males = 235), against the 470 spoken sentences uttered by ten male/female speakers which were used in [5].

Figure 9 shows the comparison of our performance with the ones in the previous work [5]. From comparing our best retrieval performances (i.e., "SVM-redundant" in each vocabulary size) with [5]'s results in each evaluation method, [5]'s results are better than ours from using the LM with 20,000 vocabularies. This is explained by the fact that the recognition performance for female speakers in [5] is better than ours for only male speakers as shown in Fig. 10[†]. Nevertheless from using the LM with 60,000 vocabularies, we consider that our best result outperformed those of Fujii et al. even through the recognition performance of female speakers is better than that for male speakers [14]. This is completely different from the results in the case of 20,000. However, the model combination by SVM greatly contributes to the improvement of the retrieval accuracy by enhancing the performance of each LVCSR model where the larger size of vocabulary is used.

6. Conclusion

In the present study, the techniques of combining outputs of multiple LVCSR models in recognition of spoken queries were evaluated. The retrieval accuracies in the NTCIR-3

[†]In this figure, the recognition performance shows in "WER"(word error rate, 100 - word accuracy [%]) and "TER"(term error rate, 100 - keyword accuracy [%]). Those measures were used in [5].

speech-driven Web retrieval task greatly depend on the vocabulary size of the LMs used in the LVCSR models. The techniques of multiple LVCSR model combination using the SVM learning can improve both the speech recognition and retrieval accuracies in speech-driven text retrieval. Especially, in terms of the retrieval accuracy, the LM with the 60,000 vocabulary size outperformed the one with 20,000 vocabulary size. In comparison of the SVM-redundant with the simple SVM, the SVM-redundant technique, which outputs more correct keywords than the simple SVM, improved the retrieval accuracies when the LM with 20,000 vocabulary size were used. On the other hand, with 60,000 vocabulary, the SVM-redundant's performance was similar to that of the simple SVM. Thus, we found it preferable to include as many correctly recognized keywords as possible, even if it adversely affects the keyword accuracy rate; too many keyword insertions, however, may further compromise the retrieval accuracies. Applying the SVM-redundant used in this paper to the speech-driven web retrieval task must be effective in improving the retrieval performance, if a suitable threshold can be found.

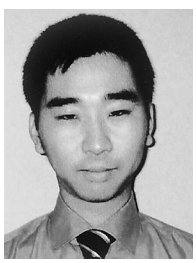
Hence, in the future works, we would like to introduce a term (keyword) weighting schema, by which only correctly recognized keywords among the large number of keyword candidates are given higher weight, for web retrieving.

Acknowledgements

We would like to thank Dr. Atsushi Fujii (University of Tsukuba) for providing the WEB page data of the NTCIR-3 and his search engine. We would also like to thank Dr. Katsunobu Itoh (Nagoya University) for providing the trigram and bigram language models with 60,000 vocabulary size.

References

[1] J.S. Garofolo, E.M. Voorhees, V.M. Stanford, and K.S. Jones, "TREC-6 1997 spoken document retrieval track overview and results," Proc. 6th Text REtrieval Conference, pp.83–91, 1997.
[2] J. Barnett, S. Anderson, J. Broglio, M. Singh, R. Hudson, and S.W. Kuo, "Experiments in spoken queries for document retrieval," Proc. EUROSPEECH'97, pp.1323–1326, 1997.
[3] F. Crestani, "Word recognition errors and relevance feedback in spoken query processing," Proc. Fourth International Conference on Flexible Query Answering Systems, pp.267–281, 2000.
[4] K. Eguchi, K. Oyama, E. Ishida, and K. Kuriyama, "Overview of Web retrieval task at the third NTCIR workshop," Working Notes of the 3rd NTCIR Workshop Meeting, pp.1–24, 2002.
[5] A. Fujii and K. Itoh, "Building a test collection for speech-driven web retrieval," Proc. EUROSPEECH2003, pp.1153–1156, 2003.
[6] T. Utsuro, Y. Kodama, T. Watanabe, H. Nishizaki, and S. Nakagawa, "Confidence of agreement among multiple lvcsr models and model combination by SVM," Proc. ICASSP2003, vol.1, pp.16–19, 2003.
[7] T. Utsuro, Y. Kodama, T. Watanabe, H. Nishizaki, and S. Nakagawa, "An emprical study on mulitple LVCSR model combination by machine learning," Proc. HTL/NAACL2004, vol.2, pp.13–16, 2004.
[8] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag, 1995.
[9] J.G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER)," Proc. ASRU, pp.347–354, 1997.
[10] H. Schwenk and J. Gauvain, "Combining multiple speech recognizers using voting and language model information," Proc. ICSLP2000, pp.915–918, 2000.
[11] V. Goel, S. Kumar, and W. Byrne, "Segmental minimum Bayes-risk ASR voting strategies," Proc. ICSLP2000, pp.139–142, 2000.
[12] G. Evermann and P. Woodland, "Posterior probability decoding, confidence estimation and system combination," Proc. NIST Speech Transcription Workshop, 2000.
[13] M. Matsushita, H. Nishizaki, T. Utsuro, Y. Kodama, and S. Nakagawa, "Evaluating multiple LVCSR model combination in NTCIR-3 speech-driven web retrieval task," Proc. EUROSPEECH2003, pp.1205–1208, 2003.
[14] T. Kawahara, T. Kobayashi, K. Takeda, N. Minematsu, K. Itoh, M. Yamamoto, A. amoto, T. Utsuro, and K. Shikano, "Sharable software repository for japanese large vocabulary continuous speech recognition," Proc. ICSLP'98, pp.763–766, 1998.
[15] A. Kai, Y. Hirose, and S. Nakagawa, "Dealing with out-of-vocabulary words and speech disfluencies in an N-gram based speech understanding system," Proc. ICSLP'98, pp.2427–2430, 1998.
[16] N. Kitaoka, N. Takahashi, and S. Nakagawa, "Large vocabulary continuous speech recognition using linear lexicon speaker with n-best approximation and tree lexicon search with 1-best approximation," IEICE Trans. Inf. & Syst. (Japanese Edition), vol.J87-D-II, no.3, pp.799–807, March 2004.
[17] S. Nakagawa and K. Yamamoto, "Evaluation of segmental unit input HMM," Proc. ICASSP'96, pp.439–442, 1996.
[18] S. Nakagawa, Speech Recognition Based on Stocastic Model, IEICE, 1988.
[19] S. Nakagawa, Y. Hirata, and Y. Hashimoto, "Japanese phoneme recognition using continuous parameter hidden markov models," J. Acoust. Soc. Japan, vol.46, no.9, pp.486–496, 1990 (in Japanese).



Masahiko Matsushita was born in 1979. He received his B.E. and M.E degrees in information and computer sciences from Toyohashi University of Technology in 2002, 2004, and now works at the Denso Techno Corporation in Kariya.

Hiromitsu Nishizaki was born in 1975. He received his B.E., M.E., and D.Eng. degrees in information and computer sciences from Toyohashi University of Technology in 1998, 2000, and 2003. He is now a research associate in the Interdisciplinary Graduate School of Medicine and Engineering at University of Yamanashi. His research interests include spoken/natural language processing.

Takehito Utsuro received his B.E., M.E., and D.Eng. degrees in electrical engineering from Kyoto University in 1989, 1991, and 1994. After serving at Nara Institute of Science and Technology and Toyohashi University of Technology, he has been a lecturer in the Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University, since 2003. He was a visiting scholar in the Department of Computer Science at Johns Hopkins University in 1999–2000. His professional interests in natural language processing, spoken language processing, machine learning, and artificial intelligence.

Seiichi Nakagawa received his B.E., M.E. degrees from Kyoto Institute of Technology in 1971 and 1973, and D.Eng. degrees from Kyoto University in 1977. He has been a professor in the Department of Information and Computer Sciences at Toyohashi University of Technology since 1990. He was a visiting scientist in the Department of Computer Science at Carnegie-Mellon University in 1985–1986. He received 1997 and 2001 Paper Awards from IEICE and the 1988 JC Bose Memorial Award from the Institution of Electronics Telecommunication Engineers. His major research interests include automatic speech recognition/speech processing, natural language processing, human interface, and artificial intelligence.