Evaluating Multiple LVCSR Model Combination in NTCIR-3 Speech-Driven Web Retrieval Task

Masahiko Matsushita†, *Hiromitsu Nishizaki*†, *Takehito Utsuro*‡, *Yasuhiro Kodama*†, *Seiichi Nakagawa*†

†Dpt. Information and Computer Sciences, Toyohashi University of Technology
‡Dpt. Intelligence Science and Technology, Graduate School of Informatics, Kyoto University
{masahiko,nisizaki,kodama,nakagawa }@slp.ics.tut.ac.jp, utsuro@i.kyoto-u.ac.jp

Abstract

This paper studies speech-driven Web retrieval models which accepts 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 evaluate the techniques of combining outputs of multiple LVCSR models in recognition of spoken queries. As model combination techniques, we compare the SVM learning technique and conventional voting schemes such as ROVER. We show that the techniques of multiple LVCSR model combination can achieve improvement both in speech recognition and retrieval accuracies in speech-driven text retrieval. We also show that model combination by SVM learning outperforms conventional voting schemes both in speech recognition and retrieval accuracies.

1. Introduction

Automatic speech recognition, which decodes human voice to generate transcriptions, has of late become a practical technology. It is feasible that speech recognition is 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, written queries are mainly used to search speech archives for relevant speech information. In previous works on speech-driven retrieval, on the other hand, spoken queries are 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.

Unlike those previous approaches, Fujii et al. [4] 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 and to recognize out-of-vocabulary words for speech recognition. Along with the NTCIR-3 [5] Web retrieval main task, which was organized to promote conventional text-based retrieval, they organized the "speech-driven retrieval" subtask. Fujii et al. [4] produced a reusable test collection for experiments of Web retrieval driven by spoken queries.

For the purpose of further improving 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] in 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 [7] and conventional voting schemes such as ROVER (*Recognizer output voting error reduction*) [8, 9, 10, 11].

Figure 1 illustrates the overall framework of our speech-driven text retrieval based on multiple LVCSR model combination. Query utterances are transcribed by each of multiple LVCSR models individually, and their outputs are combined by the model combination module. After excluding stopwords from the outputs of model combination module, the text retrieval module searches a target IR collection for documents relevant to the queries. As individual LVCSR models, we evaluated eight models that differ in their decoders as well as their acoustic models, while their language models are the same. We borrowed Fujii et al. [4]'s language model and the text retrieval module in the overall framework of Figure 1, where the language model was trained using the text of the target IR collection.

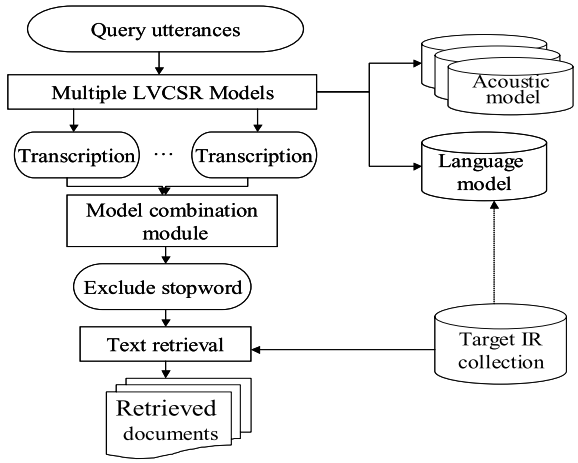

Figure 1: Speech-driven Text Retrieval based on Multiple LVCSR Model Combination

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. We also show that model combination by SVM learning outperforms conventional voting schemes such as ROVER both in speech recognition and retrieval accuracies.

2. Specification of Japanese LVCSR Models

2.1. Decoders

As the decoders of Japanese LVCSR systems, we use the one named Julius, which is provided by IPA Japanese dictation free software project [12], as well as the one named SPOJUS [13], which has been developed in our laboratory. 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 of Japanese LVCSR systems are based on Gaussian mixture HMM. We evaluate phoneme-based HMMs as well as syllable-based HMMs.

2.2.1. Acoustic Models with the Decoder JULIUS

As the acoustic models used with the decoder Julius, 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 model is gender-dependent (male).

2.2.2. Acoustic Models with the Decoder SPOJUS

The acoustic models used with the decoder SPOJUS are based on syllable HMMs, which have been developed in our laboratory [14]. The acoustic models are gender-dependent (male) syllable unit HMMs. We evaluated four types of HMMs which differ in feature parameters and/or self loop transition / duration control.

2.3. Language Model

The language model is borrowed from Fujii et al. [4], which was trained using the text of the target IR collection. From the 100GB collection of target Web text, 20,000 high-frequent words are independently used to produce word-based trigram model. The "ChaSen" (`http://chasen.aist-nara.ac.jp`) Japanese morphological analyzer was employed to extract words from the 100GB 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 10GB and the 100GB collections. In this paper, we used the 100GB collection only. The 100GB collection includes approximately 10,000,000 documents.

Ten speakers (five adult males/females) were asked to dictate 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 of 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 100GB collection, were used for evaluating both speech recognition and retrieval accuracies. Word correct and accuracy rates of the individual eight LVCSR models, averaged over the five speakers, are summarized below:

decoder	word correct (%)	word accuracy (%)
Julius	86.9(max) to 73.1(min)	78.4(max) to 66.9(min)
SPOJUS	85.0(max) to 81.8(min)	76.5(max) to 75.0(min)

4. Combining Outputs of Multiple LVCSR Models

4.1. Combination Methods

As techniques for combining outputs of multiple LVCSR models, we experimentally compare SVM learning [7] and conventional voting schemes of ROVER [8, 9, 10, 11]. The 52 queries are used for training the SVM models[1]. A Support Vector Machine is trained for choosing the most confident one among several hypothesized words from the outputs of the eight LVCSR models[2]. As features of the SVM learning, we use the IDs of the models which output the word, the part-of-speech of the word, and the syllable length of the word[3]. As 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 the ones with the confidence above this threshold as the result of model combination.

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 phase of model combination, 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 that are competing each other.

4.2. Word Recognition Rates of Spoken Queries

Figures 2 and 3 showed word correct/accuracy rates as well as keyword correct/accuracy rates of the 47 spoken queries, respectively, where averaged over the five speakers. Word correct/accuracy rates in Figure 2 are those for the whole sentences of the 47 spoken queries, while keyword correct/accuracy rates in Figure 3 are those after removing stopwords from the speech recognition outputs. Correct/accuracy rates indicated as "Julius" and "SPOJUS" are the best performing results for

[1] In this paper, an SVM model is trained using queries dictated by a single speaker and is evaluated against test queries dictated by the same speaker who dictated the training queries. We are now evaluating the performance of cross speaker SVM model combination, i.e., an SVM model for model combination is evaluated against test queries dictated by a speaker who is not a speaker of the training queries. In our previous work [6], SVM model combination was evaluated in cross speaker model combination and performed quite well.

[2] We used SVM^{light} (`http://www.cs.cornell.edu/People/tj/svm_light/`) as a tool for SVM learning.

[3] 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.

each of the two decoders. As the recognition rates for the conventional voting schemes of ROVER, "Weighted Majority Vote" shows the performance when the word correct rate of each *sentence* is used as the weight of hypothesized words, where the word correct rate of each sentence are simply estimated by linearly transforming its sentence score into word correct rate. "Majority Vote" shows the performance of the strategy of outputting no word at a tie in its voting scheme. Finally, "All_or correct" shows the performance of taking the union of all the correctly recognized words from the outputs of the eight LVCSR models without including any of recognition error words. These performance of "All_or correct" corresponds to the upper bounds of the approaches of combining outputs of multiple LVCSR models.

As can be clearly seen from these results, model combination techniques such as SVM models and conventional voting schemes achieved improvement in both word and keyword recognition rates. Furthermore, roughly comparing SVM models (i.e., SVM and SVM (redundant)) with the conventional voting schemes, SVM models outperformed the voting schemes. As we expected, SVM (redundant) improved word/keyword correct rates, while damaging its word/keyword accuracy rates.

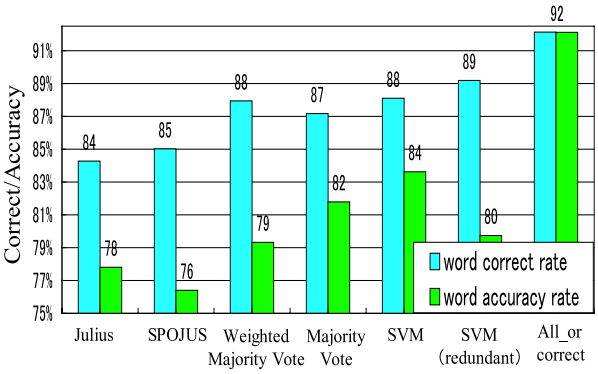

Figure 2: Word Recognition Rates of Spoken Queries

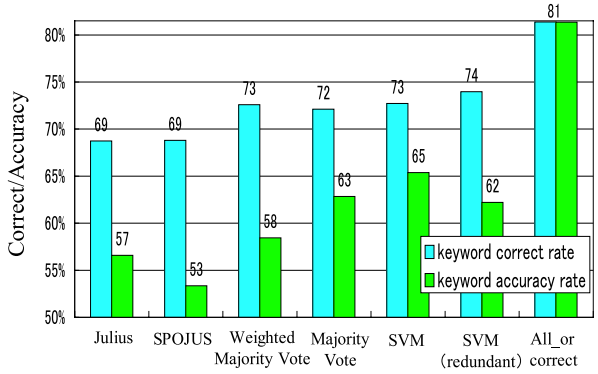

Figure 3: Keyword Recognition Rates of Spoken Queries

5. Web Retrieval

5.1. Text Retrieval Model

The text retrieval model is also borrowed from Fujii et al. [4]. It is based on an existing probabilistic retrieval method [15], 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 that 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 transcribed keywords sequence, the text retrieval module searches 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 100GB Web text collection. After excluding stopwords from the words sequence, remaining words are used as index keywords.

5.2. Evaluation Measures

Relevance assessment was 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 were optionally used for relevance assessment. To sum up, the following four assessment types were 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 investigated non-interpolated average precision values. Here, we used the 47 queries to retrieve 1,000 top documents and used the TREC evaluation software to calculate non-interpolated precision values. Finally, those average precision values are further averaged over the five speakers.

5.3. Evaluation Results

Figure 4 compares Web retrieval performance between individual LVCSR models and model combination methods. Results for Julius and SPOJUS are the best performing ones for each of the two decoders. Unexpectedly, the best performance for SPOJUS is over that for Julius, which is the opposite to 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. Web retrieval performance of the voting schemes are slightly better than the best performance for Julius, but quite close to that for SPOJUS. Web retrieval performance of the SVM models (i.e., SVM and SVM (redundant)) are mostly significantly better than those of the individual LVCSR models and the voting schemes. Comparing

the SVM and the SVM (redundant), the latter outperforms the former, indicating that it is better to include as many correctly recognized keywords as possible, even if it damages keyword accuracy rates. It is interesting to see that 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 recognition error keywords than the SVM, it seems difficult to improve the performance when (highly) relevant documents are required to be retrieved.

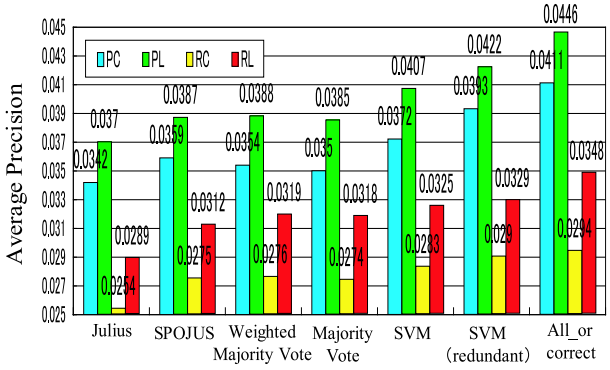

Figure 4: Comparison of Web Retrieval Performance among Model Combination Methods

Figure 5 compares Web retrieval performance between spoken queries (those indicated as SVM (redundant) and "All_or correct") and text queries (those indicated as "Text"). Considering the fact that the keyword correct rate of "All_or correct" is 81% in Figure 3 and that of "Text" is 100%, it is very surprising to see the huge gaps of their retrieval performance in Figure 5. Those huge gaps are mainly explained by the difficulty of the Web retrieval task with 100GB Web text collection. The target IR 100GB collection (about 10,000,000 documents) is huge, while the number of documents relevant to a query is very small. Therefore, removing about 20% of keywords in a query causes severe drops in the retrieval performance.

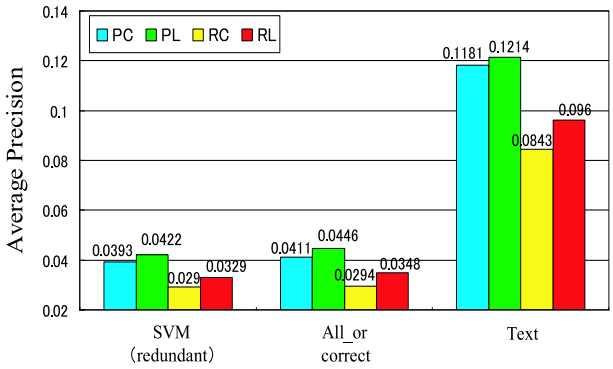

Figure 5: Comparison of Web Retrieval Performance between Spoken/Text Queries

In order to overcome those huge gaps between "All_or correct" and "Text", first of all, it is necessary to improve the keyword correct rate of "All_or correct". In the experimental results reported in this paper, we only used the first best hypothesis from each of the individual LVCSR models, and discarded all the other hypotheses with less scores. So, first, it may be useful to examine less confident hypotheses and to explore whether it is possible to improve the keyword correct rate of "All_or correct". Next, for the purpose of selectively outputting keywords that are useful in text retrieval task and discarding other less useful words, it should be quite promising to consider the keyword weights computed in the query/document similarity $sim(Q, D_i)$ in the framework of LVCSR model combination based on SVM learning. We are now working on formalizing SVM model training so that it can measure the confidence of keywords based not only on their correct/accuracy rates of speech recognition, but also on their usefulness in the text retrieval task.

6. Conclusion

This paper evaluated the techniques of combining outputs of multiple LVCSR models [6] in recognition of spoken queries of the NTCIR-3 speech-driven Web retrieval task. The techniques of multiple LVCSR model combination can achieve improvement both in speech recognition and retrieval accuracies in speech-driven text retrieval. Model combination by SVM learning outperformed conventional voting schemes both in speech recognition and retrieval accuracies.

7. 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," in *Proc. 6th Text REtrieval Conference*, 1997, pp. 83–91.

[2] J. Barnett, S. Anderson, J. Broglio, M. Singh, R. Hudson, and S. W. Kuo, "Experiments in spoken queries for document retrieval," in *Proc. 5th Eurospeech*, 1997, pp. 1323–1326.

[3] F. Crestani, "Word recognition errors and relevance feedback in spoken query processing," in *Proc. Fourth International Conference on Flexible Query Answering Systems*, 2000, pp. 267–281.

[4] A. Fujii and K. Itou, "Evaluating speech-driven web retrieval in the third NTCIR workshop," in *Proc. AAAI Spring Symposium: Intelligent Multimedia Knowledge Management*, 2003.

[5] *Proc. Third NTCIR Workshop Meeting*. National Institute of Informatics, Japan, 2002.

[6] T. Utsuro, Y. Kodama, T. Watanabe, H. Nishizaki, and S. Nakagawa, "Confidence of agreement among multiple LVCSR models and model combination by SVM," in *Proc. 28th ICASSP*, 2003, (to appear).

[7] V. N. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, 1995.

[8] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER)," in *Proc. ASRU*, 1997, pp. 347–354.

[9] H. Schwenk and J.-L. Gauvain, "Combining multiple speech recognizers using voting and language model information," in *Proc. 6th ICSLP*, 2000, vol. II, pp. 915–918.

[10] V. Goel, S. Kumar, and W. Byrne, "Segmental minimum Bayes-risk ASR voting strategies," in *Proc. 6th ICSLP*, 2000, pp. 139–142.

[11] G. Evermann and P. Woodland, "Posterior probability decoding, confidence estimation and system combination," in *Proc. NIST Speech Transcription Workshop*, 2000.

[12] T. Kawahara et al., "Sharable software repository for Japanese large vocabulary continuous speech recognition," in *Proc. 5th ICSLP*, 1998, pp. 3257–3260.

[13] A. Kai, Y. Hirose, and S. Nakagawa, "Dealing with out-of-vocabulary words and speech disfluencies in an n-gram based speech understanding system," in *Proc. 5th ICSLP*, 1998, pp. 2427–2430.

[14] S. Nakagawa and K. Yamamoto, "Evaluation of segmental unit input HMM," in *Proc. 21st ICASSP*, 1996, pp. 439–442.

[15] S. Robertson and S. Walker, "Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval," in *Proc. 17th SIGIR*, 1994, pp. 232–241.