論文 / 著書情報
Article / Book Information

Title	Robust spoken term detection using combination of phone-based and word-based recognition
Authors	Kenji Iwata, Koichi Shinoda, SADAOKI FURUI
Citation	Proc. INTERSPEECH2008, Vol. , No. , pp. 2195-2198,
Pub. date	2008, 9
Copyright	(c) 2008 International Speech Communication Association, ISCA
DOI	http://dx.doi.org/

interspeech 2008 incorporating SST 08

Robust Spoken Term Detection Using Combination of Phone-Based and Word-Based Recognition

Kenji Iwata, Koichi Shinoda, and Sadaoki Furui

Department of Computer Science, Tokyo Institute of Technology, Japan
iwata@ks.cs.titech.ac.jp, {shinoda, furui}@cs.titech.ac.jp

Abstract

We propose a robust spoken term detection method against word recognition errors using a combination of phone-based and word-based recognition. Conventional methods based on similar frameworks are problematic because phone-based recognition produces a large number of insertion errors. In our method, different substitution penalties are assigned for phone pairs to reduce such errors. We evaluated our method using the corpus of spontaneous Japanese. When recall was fixed at 50%, precision improved to 4.4 points above detection using only word-based recognition. We also report here on the effectiveness of optimization of the combination weight for each keyword.

Index Terms: Spoken term detection, Phone recognition, Word recognition

1. Introduction

In recent years, the amount of multimedia data has been rapidly increasing. User's demand for effective information retrieval is also increasing. Effective multimedia information retrieval, not only of text, but also of sound, image, and video, is in strong demand.

In this paper, we focus on spoken term detection from speech data, where each query consists of one keyword. In this simple framework, user can identify the times at which a keyword occurs through all the speech data. We use lecture speech data as examples of speech data. Since lecture speech data are very spontaneous, more recognition errors occur than during the read speech, such as those in news programs. We thus need to find a robust keyword detection method against recognition errors. To do this, a combination of sub-word based recognition and word-based recognition has been extensively studied[1, 2]. Logan *et al.* used sub-word units generated from phone expressions and was in favor of a combination of word and sub-word level indexing[1]. Yu *et al.* proposed a method using a combination of word-based and phone-based recognition[2]. They also proposed a two-pass system in which an approximate match is first executed on an entire set of documents to produce a small collection of documents. A costly detailed phonetic match is then executed for a few documents in the collection. In their work, time alignment between a phone lattice and a phone sequence of a keyword was executed to detect the frame periods of keyword utterances from phone-based recognition results[3]. These studies combining phone-based recognition with word-based recognition mainly focused on detecting out-of-vocabulary (OOV) keywords and did not aim at improving detection performance in general. This is because phone-based recognition produces a large number of insertion errors. To improve phone-based recognition, we assign different substitution penalties for phone pairs. We also combine phone-based and word-based detection to improve speech detection performance. We carry out our detection experiments under the condition that there is no OOV keyword to confirm the effectiveness of our method.

Word recognition accuracies for keywords may differ - various keywords are easily recognized, but others are not. In the latter case, phone-based recognition plays a more important role in our framework. We therefore discuss optimizing the combination weight for each keyword.

This paper is organized as follows. In Section 2, we introduce a conventional phone-based detection method. In Section 3, we explain our method for reducing insertion errors. In Section 4, we explain the fusion method, and in Section 5 we discuss the effectiveness of the proposed method. Finally, Section 6 concludes this paper.

2. Detection algorithm using phone-based recognition

First, we explain the detection algorithm using phone-based recognition proposed by Young *et al.*[3], which we use as a baseline phone-based recognition algorithm. This algorithm calculates confidence measure if a keyword appears from the frame period of a phone lattice. This calculation is made using time alignment between the frame period of the phone lattice and the keyword phone sequence.

Let the phone sequence of a keyword be $p_1 \dots p_N$, where N is the number of phones in the keyword. In executing the time alignment, the cumulative function $C(i,e)$ is prepared. This function returns the best path score for a fraction of keyword phone sequence $p_1 \dots p_i (i \leq N)$ with end time e. With this function, frame-based time alignment can be defined as:

$$\forall t\, C(0,t) = 0 \quad (1)$$

$$C(i,e) = \max_b \begin{cases} C(i-1,b) + V(b,e,p_i) \\ C(i-1,b) + (e-b)P_s \\ \quad + \max_z V(b,e,z) \\ C(i,b) + (e-b)P_i \\ C(i-1,e) + P_d \end{cases}, \quad (2)$$

where $V(b,e,p)$ is the log-likelihood score for an arc with its beginning frame b, its end frame e, and its phone hypothesis p. Penalties for phone substitution, insertion, and deletion are P_s, P_i, and P_d, respectively, and $C(N,t)$ is defined as the likelihood of the phone sequence ending at time t.

Next, a confidence measure for each phone sequence is calculated. The confidence measure of a keyword hypothesis with its beginning frame b and its ending frame e is calculated as:

$$P_I = \frac{10^{C(N,e)}}{\mathrm{ML}(b,e)}, \quad (3)$$

where $\mathrm{ML}(b,e)$ is part of the maximum likelihood score from

Accepted after peer review of full paper

frame b to e. Hypotheses are ordered by the confidence measure of the sequence.

3. Reducing insertion errors

The phone-based detection algorithm explained in Section 2 tends to produce a large number of insertion errors. We propose a method to decrease them.

3.1. Different substitution penalty using KLD

In the algorithm explained in the previous section, phone substitutions are permitted for only a limited number of phone pairs. If we further assign different substitution penalties for different phone pairs, we expect an improvement in phoneme recognition performance. Here, we determine the substitution penalty for each phone pair by using Kullback-Leibler divergence (KLD) between the two phone models[4].

First, for KLD between two GMMs s, $\tilde{s}$ is approximated using unscented transform[5] as [6]:

$$D_{\mathrm{KL}}(s||\tilde{s}) \approx \frac{1}{2L}\sum_{m=1}^{M}\omega_m \sum_{k=1}^{2L}\log\frac{p(\boldsymbol{o}_{m,k}|s)}{p(\boldsymbol{o}_{m,k}|\tilde{s})}, \qquad (4)$$

where L is the dimension of an acoustic feature vector, M is the number of mixtures, ω_m is the mixture weight of the m-th Gaussian component in the GMM, and $\boldsymbol{o}_{m,k}(1 \le k \le 2N)$ is the k-th sigma point of the m-th Gaussian component. We select $\boldsymbol{o}_{m,k} = \boldsymbol{\mu}_m + \sqrt{N\lambda_{m,k}}\boldsymbol{u}_{m,k}$, $\boldsymbol{o}_{m,k+N} = \boldsymbol{\mu}_m - \sqrt{N\lambda_{m,k}}\boldsymbol{u}_{m,k}(1 \le k \le N)$ as the sigma point, where $\boldsymbol{\mu}_m$ is the mean vector of the m-th Gaussian component, $\lambda_{m,k}$, and $\boldsymbol{u}_{m,k}$ is the k-th eigenvalue and eigenvector of the covariance diagonal matrix of the m-th Gaussian component, respectively. Note that KLD is asymmetric; for example, $D_{\mathrm{KL}}(s||\tilde{s}) \neq D_{\mathrm{KL}}(\tilde{s}||s)$.

After KLDs between all GMMs are calculated, the phone KLD for each phone pair is approximated by using dynamic programming. Using this KLD, the substitution term $P'_s(a,b)$ from phone a to phone b is expressed as:

$$\begin{aligned} P'_s(a,b) &= C(i-1,b) + (e-b)P_s \cdot \mathrm{KLD}(a||b) \\ &\quad + \max_z V(b,e,z). \end{aligned} \qquad (5)$$

3.2. Reducing number of hypotheses

The phone-based detection algorithm explained above produces a large number of hypotheses. We reduce the number of hypotheses by selecting the hypothesis with the largest value of confidence measure among hypotheses with similar frame periods. Defining whether a pair of hypotheses is similar from the viewpoint of the frame period is needed. Here, two hypotheses A and B are defined as similar if A and B satisfy:

$$\frac{\min(e_A,e_B) - \max(b_A,b_B)}{\max(e_A,e_B) - \min(b_A,b_B)} > \tau_d, \qquad (6)$$

where b_A and e_A are the beginning and end frames of A, b_B and e_B are the beginning and end frames of B, and τ_d is the threshold for the selection. A smaller τ_d reduces the number of hypotheses.

4. Combination of word- and phone-based approaches

By allowing a few recognition errors, phone-based detection is expected to detect the correct frame period for a keyword, whereas word-based detection cannot detect such a period. On the other hand, phone-based detection produces significantly more insertion errors than word-based detection. We thus combine phone- and word-based detection to bolster detection performance.

4.1. Word-based detection

To retrieve a keyword using word-based recognition, first, the 1-best word recognition results with beginning and end frames from time-alignment information are stored. Next, if the keyword is present in the 1-best results, the frame periods corresponding to the keyword are stored with the information about their frame periods.

To calculate the confidence measure of each word hypothesis, we use a likelihood average of the corresponding frame periods. Let the maximum of this averaged likelihood in all the word hypotheses be $l_{\max}$, and the likelihood of word hypothesis I be l_I. The confidence measure for I, W_I is then expressed as:

$$W_I = \frac{l_I}{l_{\max}}. \qquad (7)$$

4.2. Fusion method

We combine word- and phone-based detection results by using the confidence measures. If hypothesis I obtained by word-based detection and hypothesis J obtained by phone-based detection are satisfied by:

$$\frac{\min(e_I,e_J) - \max(b_I,b_J)}{\max(e_I,e_J) - \min(b_I,b_J)} > \tau_m, \qquad (8)$$

where b_I and e_I are the beginning and end frames of word hypothesis I, b_J and e_J are the beginning and end frames of phone sequence hypothesis J, and τ_m is the threshold, then I and J are combined to one hypothesis K. The frame period of K is the average of I and J, and the confidence measure for this combination is calculated by:

$$M_K = {W_I}^{\lambda} \cdot {P_J}^{(1-\lambda)}, \qquad (9)$$

where P_J is the confidence measure of phone sequence hypothesis J calculated by Eq.3, λ is a fusion weight that satisfies $0 \le \lambda \le 1$. If a hypothesis in either detection cannot find any hypotheses in the counterpart, the confidence measure for the counterpart has a very small value, D.

5. Experiments

5.1. Experimental conditions

We evaluated the proposed method using speech data from academic and extemporaneous lecture speaking data from the corpus of spontaneous Japanese (CSJ) [7, 8]. The number of lectures is 2701, and the total length of the data is about 530 hours. The database was divided into two parts to achieve a good balance of the same kind of lectures. We applied cross-validation using these two databases for our evaluation. The triphone acoustic model was constructed by using a training set. The language model for phone-based recognition was a simple grammar representing phone connectability in Japanese (Figure 1), and the language model for word-based recognition was a trigram constructed using the training set. We used HTK[9] for phone-based recognition and Julius[10] for word-based recognition. The model and decoder used for each recognition are shown in Table 1.

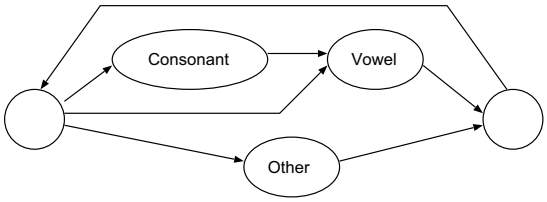

Figure 1: The grammar network used in phoneme recognition

Table 1: Model and decoder used for each recognition

Recognition Type	Word	Phoneme
Acoustic Model	Triphone	
Language Model	Trigram	Simple Grammar
Recognition Decoder	Julius	HTK

In our experiments, we selected keywords for our evaluation in the following way. We first calculated tf-idf[11] of all the words in the database, and then words with higher tf-idf than a predetermined threshold were selected as keywords. Then, to increase the reliability of the results, we excluded those words having a number of occurrences less than 100, and we further excluded particles, auxiliary verbs, and fillers. We prepared two kinds of evaluation sets. Set A was relatively small; the test set consisted of those lectures in the CSJ test set, which consists of 31 lectures. The keywords were those with the 100 largest tf-idf. Set B was large; the other half of the training set in the two-fold cross validation was used. One consisted of 1350 lectures, the other consisted of 1351 lectures. The keywords were those with the 1000 largest tf-idf. It should be noted there were no OOV keywords in our experiments.

The "correct" frame period for each keyword used in our evaluation was generated using Viterbi forced alignment. In our evaluation, if the frame period of a hypothesis and the correct frame period were superposed, even if only slightly, detection was regarded as successful. We evaluated the detection results in terms of mean average precision (MAP), precision, and recall. Mean average precision was calculated using the mean of an average precision for each keyword, and the average precision AP_I for keyword I was calculated as:

$$\mathrm{AP}_I = \frac{1}{N_I} \sum_{k=1}^{M_I} \frac{k}{r_{I,k}}, \tag{10}$$

where N_I is the number of correct periods of keyword I, M_I is the number of correctly detected periods, and $r_{I,k}$ is the rank of the k-th detected correct period of keyword I. Mean average precision is often used in video information retrieval. In calculating precision and recall, we evaluated only those hypotheses having a confidence measure larger than threshold τ_f. By changing τ_f, the precision and the recall are controlled.

5.2. Results

5.2.1. Detection results using phone-based recognition

We first evaluated the detection method using phone-based recognition with Set A. We compared the results of the method in which different substitution penalties are used for each phone pair (KLD) with results of the method in which such penalties are not used (Baseline). In all experiments, the number of hypotheses reduced using the method explained in 3.2. Threshold τ_f used to evaluate precision and recall was 10^{-22}. The detection results are shown in Table 2. In this table, we also show the results of detection using word-based recognition. Results show that KLD failed to significantly improve performance. This is because, since the size of the generated lattice was too large and correct phones were almost always present, there were few cases in which KLD was effective. Compared with the detection results using word-based recognition, the recall of phone-based detection was higher, whereas the precision of phone-based detection was much lower. Clearly, the problem with regard to insertion errors still remained.

Table 2: Detection results using phoneme recognition

Method	MAP	Precision (%)	Recall (%)
Baseline	0.365	31.8	66.2
KLD	0.364	31.7	65.5
Word recognition	0.546	69.3	63.9

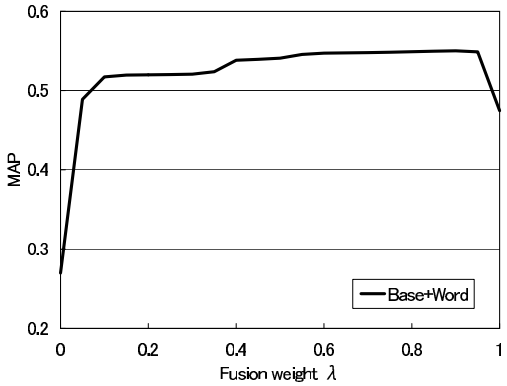

Figure 2: Fusion results of phone- and word-based detection

5.2.2. Fusion results

Next, we evaluated the results of the combination. The evaluation set was Set B. Parameters used with the combination are $D = 10^{-1000}$, $\tau_m = 0.3$. The mean average precision values with different fusion weights λ are shown in Figure 2. It should be noted that when we used $\lambda = 1$, only word-based detection was used. These results show that the combination of phone- and word-based detection significantly improved MAP. When we combine phone- and word-based detection results at $\lambda = 0.9$, MAP is the highest; 0.550 higher than that of the word-based detection results, 0.475.

A precision-recall curve when MAP is the highest is shown in Figure 3. The optimal value of λ for Base+Word was 0.9. When recall was fixed at 50%, precision improved by 4.4 points for detection using word-based recognition only. This shows that the combination method effectively raised the ranks of the correct hypotheses.

Several previous studies reported that the combination of phone- and word-based detection did not outperform word-based detection when only in-vocabulary keywords were used (e.g. [1]). In our experiments, the size of the phone lattice constructed for each utterance was very large; the average branching factor for each node was about five. Therefore, our method

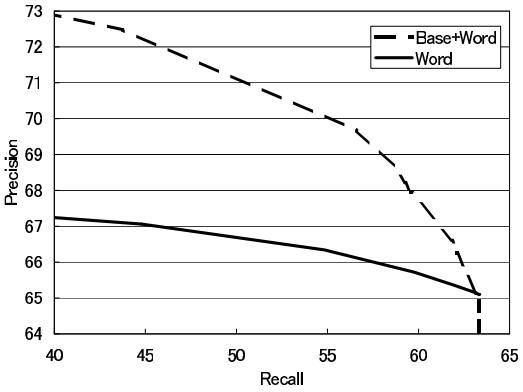

Figure 3: Precision-recall curve of fusion results

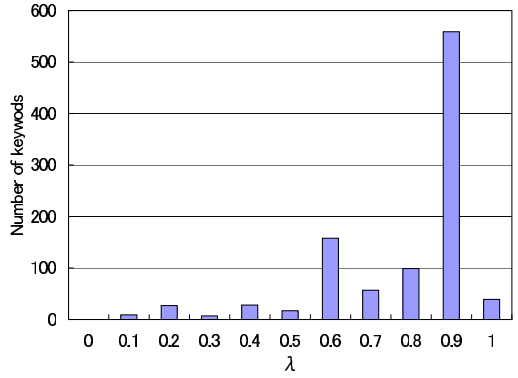

Figure 4: Relationship between fusion weight λ and number of keywords

was able to find the correct hypothesis in the phone lattice even when the word-based recognition failed. This might be one of the reasons that our method had better accuracies with the combination. On the other hand, it should be noted that our method required a larger amount of computational resources than the conventional methods.

5.2.3. Weight control for each keyword

We evaluated the method, Base+Word, for each keyword. We optimized the fusion weight λ for each keyword by changing 0 to 1 by 0.1. For each value of λ, the number of keywords with an optimal average precision value is shown in Figure 4. The results show that, while most keywords had larger weights on word detection results, some keywords had larger weights on phone-based detection results. These results show that optimizing the fusion weight for each keyword is effective. Mean average precision improved to 0.555 by using optimal λ for each keyword. The F-measure also improved from 64.2 to 65.1 by using optimal λ and the confidence measure threshold τ_f for each keyword.

6. Conclusion

We have developed a spoken term detection method using a combination of phone- and word-based recognition. To reduce insertion errors, we applied this method to a detection algorithm. Assigning different substitution penalties for phone pairs in the phone lattice was introduced. We evaluated these methods using CSJ. When the recall was fixed at 50%, the precision of the combined approaches improved to 4.4 points above the precision level obtained only using word-based detection. We also discussed optimizing the fusion weight and the confidence measure threshold for each keyword.

In the future, we plan to use the phone language model for detection to improve performance and include OOV keywords for our evaluation. In addition, reducing the computational costs of the time alignment is strongly needed. For example, using Confusion Network[12], which compresses the lattice, might be effectively included without decreasing the performance of our proposed method. There is also potential in exploring whether the optimal fusion weight and confidence measure threshold can be estimated automatically.

7. References

[1] B. Logan, P. Moreno, and O. Deshmukh, "Word and subword indexing approaches for reducing the effects of OOV queries on spoken audio", Proc. Human Language Technology Conference, pp. 31-35, 2002

[2] P. Yu, K. Chen, C. Ma, and F. Seide, "Vocabulary-independent indexing of spontaneous Speech", IEEE Transactions on Speech and Audio Processing, vol. 13, no. 5, pp. 635-643, 2005

[3] D. A. James and S. J. Young, "A fast lattice-based approach to vocabulary independent wordspotting", Proc. ICASSP, pp. I-377-I-380, 1994

[4] P. Liu, F. K. Soong, and J-L. Zhou, "Divergence-based similarity measure for spoken document retrieval", Proc. ICASSP 2007, pp. IV-89-IV-92, 2007

[5] J. Goldberger, S Gordon, and H Greenspan, "An efficient image similarity measure based on approximations of KL-divergence between two gaussian mixtures", Proc. IEEE International Conference on Computer Vision 2003, pp. 370-377, 2003

[6] J. Du, P. Liu, F. K. Soong, J-L. Zhou, and R-H. Wang, "Minimum divergence based discriminative training", Proc Interspeech 2006, pp. 2410-2413, 2006

[7] K. Maekawa, H. Koiso, S. Furui, and H. Isahara, "Spontaneous speech corpus of Japanese", Proc. LREC2000, Athens, Greece, vol. 2, pp. 947-952, 2000

[8] The Corpus of Spontaneous Japanese, National Institute for Japanese Language, http://www2.kokken.go.jp/csj/public/

[9] HMM Tool Kit (HTK) (ver. 3.2), http://htk.eng.cam.ac.uk/

[10] Julius (ver. 3.5.3), http://julius.sourceforge.jp/en/julius.html

[11] Salton, G. and McGill, M. J., "Introduction to modern information retrieval", McGraw-Hill, 1983

[12] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks", Computer Speech and Language, vol. 14, no. 4, pp. 373-400, 2000