INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Generating complementary acoustic model spaces in DNN-based sequence-to-frame DTW scheme for out-of-vocabulary spoken term detection

Shi-wook Lee [1], *Kazuyo Tanaka*[2], *Yoshiaki Itoh*[3]

[1] National Institute of Advanced Industrial Science and Technology, Japan
[2] Tsukuba University, Japan
[3] Iwate Prefectural University, Japan

s.lee@aist.go.jp, tanaka.kazuyo.gb@u.tsukuba.ac.jp, y-itoh@iwate-pu.ac.jp

Abstract

This paper proposes a sequence-to-frame dynamic time warping (DTW) combination approach to improve out-of-vocabulary (OOV) spoken term detection (STD) performance gain. The goal of this paper is twofold: first, we propose a method that directly adopts the posterior probability of deep neural network (DNN) and Gaussian mixture model (GMM) as the similarity distance for sequence-to-frame DTW. Second, we investigate combinations of diverse schemes in GMM and DNN, with different subword units and acoustic models, estimate the complementarity in terms of performance gap and correlation of the combined systems, and discuss the performance gain of the combined systems. The results of evaluations conducted of the combined systems on an out-of-vocabulary spoken term detection task show that the performance gain of DNN-based systems is better than that of GMM-based systems. However, the performance gain obtained by combining DNN- and GMM-based systems is insignificant, even though DNN and GMM are highly heterogeneous. This is because the performance gap between DNN-based systems and GMM-based systems is quite large. On the other hand, score fusion of two heterogeneous subword units, triphone and sub-phonetic segments, in DNN-based systems provides significantly improved performance.

Index Terms: spoken term detection, keyword search, system combination, deep neural network, Gaussian mixture model, subword unit

1. Introduction

In the field of automatic speech recognition (ASR) and statistical machine translation, combining the outputs of diverse systems to improve performance has been extensively researched [1-16]. In ASR, systems are combined using schemes such as ROVER [1], confusion network combination (CNC) [2], and minimum Bayes risk (MBR) [3, 4]. It has also been reported that significant improvements on STD tasks can be obtained by carefully selecting diverse ASR components, such as acoustic model, decoding strategy, and audio segmentation [5-7]. The complementarity of the combined systems is crucially important to performance improvement, where the systems being combined are independently trained and combined in post-processing steps [8-12]. When the performance gap is very large, the combination has often been seen to yield negligible gains and even degraded performance. Therefore, combining independent systems with comparably high performance is desirable [13, 14]. Both the performance gap and similarity of detected candidates are highly correlated with performance gain. However, the systems being combined are typically not guaranteed to be complementary and deriving a complementary system theoretically is very difficult. Niyogi et al. [14] designed multiple systems through a procedure that directly minimizes the correlation of their respective errors. Boosting is a machine learning technique that is specifically designed to generate a series of complementary systems [15, 16]. The aim of boosting is to train a number of systems that may perform poorly individually, but perform well in combination.

Spoken term detection (STD) is used to locate all occurrences of the query word/phrase in the search audio database [17, 18]. Almost all ASR systems employ a fixed vocabulary. Words that are not in this fixed vocabulary, OOV words, are not correctly recognized by the ASR system, but are instead misrecognized as an alternate with similar acoustic features. This results in the subsequent word-based STD not being properly conducted. The effects of OOV words in STD can be rectified using subword-based detection [19-23] or phonetic posteriorgram template matching [24, 25]. In subword-based STD, system combination can be carried out by score fusion of the frames or detected lists. The simplest frame-synchronous combination technique fuses the posterior probabilities of the combined systems. When the systems being combined have different frame configurations, fusing the scores of the time-equivalent ranked lists during post-processing is preferred. Subword-based STD thus benefits from combination, because combination can be carried out at various stages and on various schemes. DNN is being successfully employed in ASR nowadays [12, 26-28]. Swietojanski et al. [4] reported that combing GMM-hidden Markov model (HMM) and DNN-HMM systems with MBR-based combination of lattice leads to reduced word error rate in ASR. In this paper, we investigate the combination effect of heterogeneous systems on GMM- and DNN-based STD. We hypothesize that because DNN and GMM are highly heterogeneous combining them can yield further performance gain.

The remainder of this paper is organized as follows: Section 2 describes sequence-to-frame dynamic time warping for STD. Section 3 discusses score fusion of diverse systems. Section 4 presents the results of experimental evaluations that show that combination with a new subword unit can maximize diversity and yield better improvement than other combination approaches, which are carried out using different feature inputs and different subword units in DNN- and GMM-based systems. Finally, Section 5 concludes this paper.

http://dx.doi.org/10.21437/Interspeech.2016-838

2. Sequence-to-frame dynamic time warping for OOV STD

In sequence-to-frame DTW, a query is first transformed into one of three types of symbolic sequence representations: context-dependent phoneme, in practice simply called triphone; sub-phonetic segmentation (SPS); or their HMM state. We varied the subword based on linguistic knowledge to derive a new proposed subword unit, SPS, to alter the model space of the conventional triphone. The novel SPS combined with a triphone resulted in improved performance gain [23]. The sequence-to-frame DTW is based on the following:

$$G(q,r) = min\begin{Bmatrix} G(q,r-1) + D[(q,r)|(q,r-1)] \\ G(q-1,r-1) + D[(q,r)|(q-1,r-1)] \\ G(q-1,r) + D[(q,r)|(q-1,r)] \end{Bmatrix} \tag{1}$$

where q is an HMM-state or a subword of a subword sequence of a query, and r is a frame of the search audio database. Here, although both subwords and HMM-states of subwords are tested in experiments, for convenience, we simply denote them HMM-states. $G(q,r)$ denotes the cumulative dissimilarity of an HMM-state, q, up to the r-th frame. $G(q,r)$ is normalized in the last HMM-state of a query by the detected interval and this normalized dissimilarity value is used as *score*. The portion of the score that is less than a predefined threshold value is detected as a spoken term and ranked in a detected list. In the right side of Eq. (1), the first path corresponds to self-transition in HMM and the second path is other-transition. The third is deletion of state, where it can be expressed as skip-transition—which is not usually employed in the common 3-state HMM topology of current ASR systems. The second term on the right side of Eq. (1), $D[\cdot]$, is the sequence-to-frame dissimilarity distance. This DTW calculation is a variant of the Levenshtein distance, in which the local dissimilarity distance is practically calculated by posterior probability.

In this paper, two kinds of posterior probability are adopted for the sequence-to-frame dissimilarity distance: scaled likelihood of GMM, given in Eq. (2), and softmax output of DNN, given in Eq. (6). The posterior probability of state q given the acoustic observation $\boldsymbol{o}_t$ at frame t from the acoustic likelihood of GMM is estimated as,

$$p(q|\boldsymbol{o}_t) = \frac{p(\boldsymbol{o}_t|q)P(q)}{\sum_{q_k \in Q} p(\boldsymbol{o}_t|q_k)P(q_k)} \tag{2}$$

$$\begin{aligned} D_{GMM} &= -log\big(p(q|\boldsymbol{o}_t)\big) \\ &\approx -log\big(p(\boldsymbol{o}_t|q)\big) + \log\left(\sum_{q_k \in Q} p(\boldsymbol{o}_t|q_k)\right) \end{aligned} \tag{3}$$

Using noninformative priors, uniform distribution $P(q_k) = const. \forall q_k \in Q$, and taking negative logarithm from the scaled likelihood of Eq. (2), the local dissimilarity distance of GMM is the negative log state posterior probability, Eq. (3).

A DNN, as used in this paper to calculate the HMM-state posterior probability, $p(q|\boldsymbol{o}_t)$, is a feed-forward, artificial neural network from a stack of $(L+1)$ layers, where $(L-1)$ hidden layers are log-linear models between the *0*-th input layer and the top *L*-th output layer [26]. Each hidden unit, *j*, of the l-th layer uses the logistic function to map its total input, x_j, from the $(l-1)$-th layer into the scalar state, y_j, that it sends to the l-th layer.

$$x_j = b_j + \sum_i y_i w_{ij} \tag{4}$$

$$y_j = logistic(x_j) = \frac{1}{1 + exp(-x_j)} \tag{5}$$

where b_j is the bias of unit j, i is an index over units in the $(l-1)$-th layer, and w_{ij} is the weight on a connection to unit j from unit i in the $(l-1)$ layer. For state posterior probability, $p(q|\boldsymbol{o}_t)$, each unit j of the top *L*-th output layer converts its total input, $x_j = x_q^L$, using the softmax function as follows:

$$p(q|\boldsymbol{o}_t) = \frac{exp(x_q^L)}{\sum_{q_k \in Q} exp(x_{q_k}^L)} \tag{6}$$

$$\begin{aligned} D_{DNN} &= -log\big(p(q|\boldsymbol{o}_t)\big) \\ &= -x_q^L + \log\left(\sum_{q_k \in Q} exp(x_{q_k}^L)\right) \end{aligned} \tag{7}$$

Further, the local dissimilarity distance of DNN is calculated in Eq. (7) by taking the negative logarithm of the state posterior probability of Eq. (6).

3. Score fusion of complementary systems

We surmise that combining detection candidates generated by different systems can yield performance gain over all individual systems. Score fusion of systems can be performed at various levels—frame, state, or detected term. The simplest approach is to perform frame-synchronous combination by using a linear interpolation of the observation log-likelihoods of *N* multiple systems as

$$log\, p(\boldsymbol{o}_t|q) = \sum_{n=1}^{N} \alpha_n \, log\, p_n(\boldsymbol{o}_t|q), where \sum_{n=1}^{N} \alpha_n = 1 \tag{8}$$

where α_n is the interpolation weight of system n, $p(\boldsymbol{o}_t|q)$ is the combined likelihood of observation $\boldsymbol{o}_t$ given the HMM-state q, and $p_n(\boldsymbol{o}_t|q)$ is the likelihood from the *n*-th system [4, 12].

In order to apply unified score fusion for various frame configurations, HMM-state of GMM-based systems and input and output layers of DNN-based systems, we perform score fusion on detected term lists at the final detection decision. First, the detected term lists are aligned across systems based on the overlap of timespans, and the score of the aligned terms are fused across the N systems as,

$$\hat{s}_d = \sum_{n=1}^{N} \alpha_n \cdot s_{d,n}, \qquad where \quad \sum_{n=1}^{N} \alpha_n = 1 \tag{9}$$

where d is the overlapped alignment term which is the detection result given by ranking the similarity scores, n denotes the *n*-th system being combined, $s_{d,n}$ is the score of detected term d of the *n*-th system, and $\hat{s}_d$ is the merged score of detected term d. If a detected term does not appear in any system's list, that system is assumed to have assigned it zero probability. In experiments, the interpolation weight α is empirically decided for best performance.

4. Experimental results

4.1. Spoken Term Detection Task

In this section, the results of experiments conducted on NTCIR10 STD task data, which are fully described in [29, 30], are presented and analyzed. The data comprise a total of 104 oral presentations (28.6 hours) for the search audio database, along with 100 queries and their relevant segments.

In the experiments, two feature vectors were extracted from 186 hours of Corpus of Spontaneous Japanese data [31]. The first feature vector for both triphone and SPS consisted of 12-dimensional Mel-frequency cepstral coefficient (MFCC) and one power with first and second derivatives—a total of 39 dimensions. The second feature vector, for DNN only, consisted of a 40-dimensional log filter-bank (FBANK) with first and second derivatives—a total of 120 dimensions. For DNN training, the input layer was formed from a context window comprising 11 frames, creating an input layer of 429 units for MFCC and 1320 units for FBANK. The DNN had one, three, and five hidden layers, each with 2048 units. The respective number of units for the output layer was 430 for SPS, 1290 for SPS-state, 10325 for triphone, 30975 for triphone-state, and 3078 for phonetic decision tree based tied triphone-state. These specifications are summarized in Table 1.

Table 1: *Summary of input layers, output layers, and respective number of units in the DNN-based systems.*

Feature of input layer	Number of units
MFCC	429
FBANK	1320
Subword or state of output layer	**Number of units**
Triphone (TRI)	10325
Triphone state (TRI-state)	30975
Tied triphone state (TiedTRI-state)	3078
SPS	430
SPS state (SPS-state)	1290

The networks were initialized using layer-by-layer generative pre-training and then discriminatively trained using backpropagation and the cross-entropy criteria. GMM with maximum likelihood estimation was used for forced alignment in DNN. DNN training was carried out using stochastic mini-batch gradient descend with a mini-batch size of 256 samples. During pre-training, a learning rate of 2.0e-3 per mini-batch was used for the first Gaussian-Bernoulli restricted Boltzmann machine (RBM) layer, a learning rate of 5.0e-3 per mini-batch for the remaining Bernoulli-Bernoulli RBM layers, and a learning rate of 8.0e-3 per mini-batch during fine-tuning.

To evaluate performance, we used *average of maximum F-measure* (AMF), which averages the maximum F-measure (harmonic mean of precision and recall) of all queries, and then multiplied the result by 100 to obtain a single value as a percentage. This calculation is described in detail in [23].

4.2. Baseline results of individual system

Table 2 shows the baseline results obtained from the GMM-based system for various mixture numbers. Because the number of states in SPS-state (1290) differs from that in TRI-state (30975), with two mixtures per state, the performance obtained using TRI-state, 60.06, was significantly better than that obtained using SPS-state, 47.56. However, as the number of mixture components increased, the performance gap is eliminated.

Table 2: *Baseline detection results for different mixture numbers per state and different subwords in GMM-based system (values shown are AMF for the NTCIR10 STD task).*

	SPS-state	TRI-state
2 mixtures	47.56	60.06
4 mixtures	62.03	63.12
8 mixtures	65.11	**64.28**
16 mixtures	**66.90**	63.82

In previous work [23], we reported on subword-based DTW, in which text query was transformed into subword sequences and search audio database was recognized into subword sequences, and then DTW was carried out on those subword sequences. In this paper, we propose sequence-to-frame DTW, as described in Section 2. The performance of STD using sequence-to-frame DTW is better than that of the previous subword-based DTW. In fact, sequence-to-frame DTW should be adopted as post-processing after a fast indexing or matching procedure because it is computationally expensive and time-consuming [32].

Table 3 presents the results obtained for the DNN-based system. Addition of more hidden layers in DNN results in improved STD performance and convergence at DNN with three or five hidden layers. Using FBANK as the input feature in the DNN-based STD system is significantly better than using MFCC over all STD schemes, by approximately five to eight points. Further, for output units, using the subword itself, such as triphone and SPS, is far worse than state-level units. When the acoustic state is mapped down to its corresponding subword label, SPS (430) and triphone (10325), the acoustic model space becomes less discriminative for classification and the distance is less accurate for DTW. The DNN-based system, 81.03, is dramatically better than the GMM-based system, 66.90, which confirms a fact that is already widely known.

Table 3: *Comparison of baseline detection results with various hidden layers and input/output schemes in DNN-based system.*

Input layer	Output layer	Hidden layer and units		
		1×2048	3×2048	5×2048
MFCC	TRI	35.06	44.82	**45.04**
	TRI-state	71.38	74.90	**75.24**
	TiedTRI-state	71.97	75.28	**75.30**
	SPS	44.06	**48.29**	45.58
	SPS-state	71.04	**73.28**	73.09
FBANK	TRI	41.52	51.34	**51.76**
	TRI-state	75.61	79.94	**80.76**
	TiedTRI-state	75.97	**80.04**	79.88
	SPS	46.02	**57.53**	57.37
	SPS-state	76.62	**81.03**	79.08

The tree-based state tying approach has been studied and developed on insufficient training data with the objective of training triphones in GMM-based systems [33-35]. Seide et al. [27] and Yu et al. [28] modeled tied triphone-state directly on DNN-based ASR systems and reported that using tied triphone-state as DNN output nodes is a critical factor in achieving the unusual accuracy improvements in [27]. And Breslin et al. [13] proposed directed decision trees for generating complementary ASR systems. Accordingly, we investigated the complementarity between tied triphone-state and not-tied triphone-state. As shown in Table 3, there are very slight differences in performance between these two triphone-states, tied (TiedTRI-state) and not-tied (TRI-state), over all schemes.

Table 4: *Experimental results for combinations of two systems: All DNNs have five hidden layers with 2048 units, except 3HL, which has three hidden layers.*

1	System #1	AMF	System #2	AMF	Performance gap	Correlation coefficient	AMF combined	Performance gain (%)
2	16mix.GMM.SPSstate	66.90	16mix.GMM.TRIstate	63.82	3.08	0.4288	70.11	**4.79**
3	FBANK.SPSstate	79.08	16mix.GMM.SPSstate	66.90	12.18	0.4710	79.50	0.53
4	FBANK.TRIstate	80.76			13.86	0.3931	81.76	1.23
5	FBANK.TiedTRIstate	79.88			12.98	0.3961	78.29	-1.99
6	FBANK.SPSstate	79.08	16mix.GMM.TRIstate	63.82	15.26	0.4281	80.24	1.46
7	FBANK.TRIstate	80.76			16.94	0.4267	79.53	-1.52
8	FBANK.TiedTRIstate	79.88			16.06	0.3532	80.00	0.15
9	3HL.FBANK.SPSstate	81.03	FBANK.SPSstate	79.08	1.95	0.8217	81.07	0.04
10	3HL.FBANK.TRIstate	79.94	FBANK.TRIstate	80.76	0.82	0.8130	81.14	0.47
11	3HL.FBANK.TiedTRIstate	80.04	FBANK.TiedTRIstate	79.88	0.16	0.7993	81.06	1.27
12	MFCC.SPSstate	73.09	FBANK.SPSstate	79.08	5.99	0.7727	78.95	-0.16
13	MFCC.TRIstate	75.24	FBANK.TRIstate	80.76	5.52	0.7784	80.43	-0.40
14	MFCC.TiedTRIstate	75.33	FBANK.TiedTRIstate	79.88	4.55	0.7626	80.57	0.86
15	FBANK.TRIstate	80.76	FBANK.TiedTRIstate	79.88	0.88	0.7380	82.06	1.60
16	FBANK.SPSstate	79.08	FBANK.TiedTRIstate	79.88	0.80	0.5557	**83.57**	**4.61**
17	FBANK.SPSstate	79.08	FBANK.TRIstate	80.76	1.68	0.5764	**84.47**	**4.59**

4.3. Systems combination results

Table 4 summarizes all results for combinations of two systems. To prove that a link exists between complementarity and performance, we estimated complementarity by using the correlation coefficient of detected terms, which is calculated as follows:

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\left(\left(\sum_{i=1}^{n}(x_i - \overline{x})^2\right)\left(\sum_{i=1}^{n}(y_i - \overline{y})^2\right)\right)^{1/2}} \quad (10)$$

where $\overline{x}$ and $\overline{y}$ are the arithmetic score means of the detected terms of the systems being combined, which is shown in the column seven in Table 4.

In Table 4, the performance gain in column nine are relative values, calculated with respect to the better AMF of the systems being combined. The second row in Table 4 shows that there is a significant performance gain, 4.79%, from the combination of two different subword units, SPS and triphone, in the GMM-based system. As discussed earlier, the false alarms generated by conventional GMM- and DNN-based systems are different and has relatively very low correlation coefficient, from 0.3532 to 0.4710. This is expected to provide the possibility of improving the overall performance by fusing the complementary detection results of GMM- and DNN-based systems. However, as shown from the third to the eighth row in Table 4, because of the large performance gap, from 12.18 to 16.94, all performance gains from the combination of GMM- and DNN-based systems are small or negligible, and sometimes degraded. From the ninth to the eleventh row, the combination is carried out between different hidden layers, three layers and five layers in the DNN-based system. As seen in the seventh column, the correlation coefficient is relatively very high owing to their dependency, which results in a small performance gain, from 0.04% to 1.27%. From the twelfth to the fourteenth row, the combination is carried out between different input features, MFCC and FBANK. Because the performance gap is marginally significant, from 4.55 to 5.99 and the correlation coefficient is also high, approximately 0.77, the performance gains are very small. From the fifteenth row, owing to their similarity with high correlation coefficient (0.7380), the combination of tied triphone-state (TiedTRI-state) and not-tied triphone-state (TRI-state) leads to a slight 1.6% performance gain. Finally, for the sixteenth and seventeenth rows, because the performance gap is small and the correlation coefficient is also comparably low, significant performance gains, 4.59% and 4.61%, can be achieved from the combination of two subword units, SPS-state and triphone-state, and SPS-state and tied triphone-state, respectively. Finally, we achieved the best performance of 84.47 AMF from the combination of SPS-state and triphone-state. In the second, sixteenth, and seventeenth rows, both in GMM- and DNN-based systems, combinations based on different subword units, SPS-state and triphone-state, lead to significant performance gain.

5. Conclusions

In this paper, we proposed a sequence-to-frame DTW and investigated combinations of diverse schemes in GMM- and DNN-based systems comprising different subwords units and acoustic models. We showed that sequence-to-frame DTW improves STD performance compared to our previous subword-based DTW. Further, the performance of DNN-based STD systems, 81.03 AMF, was found to be dramatically better than that of GMM-based STD systems, 66.90 AMF. The results of system combination experiments confirmed that combining two systems that have low correlation coefficient and low performance gap leads to high performance gain after combination. Although DNN- and GMM-based systems are highly heterogeneous, their performance gap is quite large, and the performance gain after combination is negligible. However, the combination of two heterogeneous subword units, triphone and the proposed SPS, lead to significant performance improvements both on DNN- and GMM-based systems. Thus, we empirically confirmed that the acoustic model space using the proposed SPS is complementary to widely used triphone.

6. Acknowledgements

This research is partially supported by a Grand-in-Aid for Scientific Research (C), KAKENHI Project Nos. 15K00241 and 15K00262.

7. References

[1] J.G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER)," in Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding, pp. 347-354, 1997.

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

[3] H.Xu, D.Povey, L.Mangu, and J.Zhu, "Minimum Bayes Risk Decoding and System Combination Based on a Recursion for Edit Distance," Computer Speech and Language, vol. 25, no. 4, pp. 802-828, 2011.

[4] P. Swietojanski, A. Ghoshal and S. Renals, "Revisiting hybrid and GMM-HMM system combination techniques," in Proc. of ICASSP, pp. 6744-6748, 2013.

[5] L. Mangu, H. Soltau, H.-K. Kuo, B. Kingsbury and G. Saon, "Exploiting diversity for spoken term detection," in Proc. ICASSP, pp. 8282-8286, 2013.

[6] H. Lee, Y. Zhang, E. Chuangsuwanich, and J. Glass, "Graph-based Re-ranking using Acoustic Feature Similarity between Search Results for Spoken Term Detection on Low-resource Languages," in Proc. of INTERSPEECH, pp. 2479-2483, 2014.

[7] R. W. M. Ng, C. C. Leung, T. Lee, B. Ma and H. Li, "Score fusion and calibration in multiple language detectors with large performance variation," in Proc. of ICASSP, pp. 4404-4407, 2011.

[8] C. Breslin and M.J.F. Gales,"Generating complementary systems for speech recognition," in Proc. of INTERSPEECH, pp. 525-528, 2006.

[9] L. Burget, "Measurement of Complementarity of Recognition Systems," in Proc. of 7th International Conference, Text, Speech and Dialogue, pp. 283-290, 2004.

[10] M.J.F. Gales and S.S. Airey, "Product of Gaussians for speech recognition," Computer Speech and Language, vol. 20 no. 1, pp. 22-40, January, 2006.

[11] L.K. Hansen and P. Salamon, "Neural Network Ensembles," IEEE/ACM Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 10, pp. 993-1001, Oct. 1990.

[12] D. Yu and L. Deng, "Automatic Speech Recognition: A Deep Learning Approach," Springer-Verlag London,2015.

[13] C. Breslin and M.J.F. Gales, "Directed decision trees for generating complementary systems," Speech Communication, Vol.51 No.3, pp.284-295, March, 2009.

[14] P. Niyogi, J. Pierrot, and O. Siohan, "Multiple classifiers by constrained minimization," in Proc. of ICASSP, pp. 3462-3465, 2000.

[15] Y. Freund and R.E. Schapire, "Experiments with a New Boosting Algorithm," in Proc. of ICML, pp.148-156, 1996.

[16] Y. Freund and R.E. Schapire, "A Decision-Theoretic Generalization of online learning and an application to boosting," Journal of Computer and System Sciences, vol. 55, no. 1, pp. 119-129, 1997.

[17] NIST, "The Spoken Term Detection (STD) 2006 Evaluation Plan," http://www.nist.gov/speech/tests/std/ docs/std06-evalplan-v10.pdf, 2006. (Currently not available)

[18] L. S. Lee, J. Glass, H. Y. Lee and C. A. Chan, "Spoken Content Retrieval—Beyond Cascading Speech Recognition with Text Retrieval," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 23, no. 9, pp. 1389-1420, Sept. 2015.

[19] K. Ng, "Subword-based Approaches for Spoken Document Retrieval," PhD Thesis, MIT, 2000.

[20] M. Saraclar and R. Sproat, "Lattice-Based Search for Spoken Utterance Retrieval," in Proc. of HLT-NAACL, pp. 129-136, 2004.

[21] S. Lee, K. Tanaka, and Y. Itoh, "Combining Multiple Subword Representations for Open-vocabulary Spoken Document Retrieval," in Proc. of ICASSP, pp. 505-508, 2005.

[22] P.C. Woodland, S.E. Johnson, P. Jourlin, and K. Spärck Jones, "Effects of out of vocabulary words in spoken document retrieval," in Proc. of the 23rd annual international ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 372-374, 2000.

[23] S. Lee, K. Tanaka, and Y. Itoh, "Combination of diverse subword units in spoken term detection," in Proc. of INTERSPEECH, pp. 3658-3689, 2015.

[24] Y. Zhang, J. R. Glass, "Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams," in Proc. of IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU), pp.398-403, 2009.

[25] T. J. Hazen, W. Shen and C. White, "Query-by-example spoken term detection using phonetic posteriorgram templates," in Proc. of IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU), pp.421-426, 2009.

[26] G. Hinton et al., "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups," IEEE Signal Processing Magazine, vol. 29, no. 6, pp. 82-97, Nov. 2012.

[27] F. Seide, G. Li and D. Yu, "Conversational speech transcription using context-dependent deep neural networks," in Proc. of INTERSPEECH, pp. 437-440, 2011.

[28] D. Yu, L. Deng and G. E. Dahl, "Roles of Pre-Training and Fine-Tuning in Context-Dependent DBN-HMMs for Real-World Speech Recognition," in Proc. of NIPS workshop on Deep Learning and Unsupervised Feature Learning, 2010.

[29] T. Akiba, H. Nishizaki, K. Aikawa, X. Hu, Y. Itoh, T. Kawahara, S. Nakagawa, H. Nanjo and Y. Yamashita, "Overview of the NTCIR-10 SpokenDoc-2 Task," in Proc. NTCIR Conference, pp. 573-587, 2013.

[30] J. Tejedor and et al., "Spoken term detection ALBAYZIN 2014 evaluation: Overview, systems, results, and discussion," EURASIP Journal on Audio, Speech, and Music Processing, no. 1, pp. 1-27, December 2015.

[31] K. Maekawa, "Corpus of spontaneous Japanese: Its Design and Evaluation," in Proc. of ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition (SSPR), pp. 7-12, 2003.

[32] S. Lee, K. Tanaka, and Y. Itoh, "Effective combination of heterogeneous subword-based spoken term detection systems," in Proc. of IEEE Spoken Language Technology Workshop (SLT), pp. 436-441, 2014.

[33] S.J. Young, J.J. Odell and P.C. Woodland, "Tree-based State Tying for High Accuracy Acoustic Modelling," in Proc. of Workshop on Human Language Technology, pp.307-312, 1994.

[34] B. Ramabhadran, O. Siohan, L. Mangu, M. Westphal, H. Schulz, A. Soneiro, and G. Zweig, "The IBM 2006 Speech Transcription System for European Parliamentary Speeches," in Proc. of INTERSPEECH, pp. 1225-1228, 2006.

[35] J. Huang, E. Marcheret, K. Visweswariah, V. Libal and G. Potamianos, "Detection, diarization, and transcription of far-field lecture speech," in Proc. of INTERSPEECH, pp.2161-2164, 2007.