Optimal Automatic Speech Recognition System Selection for Noisy Environments

Yuuki Tachioka and Tomohiro Narita
Information Technology R&D Center, Mitsubishi Electric Corporation, Kamakura, Kanagawa, Japan
E-mail: Tachioka.Yuki@eb.MitsubishiElectric.co.jp Tel: +81-467-41-2072

***Abstract*—To improve the performance of noisy automatic speech recognition (ASR), it is effective to prepare multiple ASR systems that can address the large varieties of noise. However, the optimal ASR system is different for each environment and mismatches between training and testing degrade ASR performance. In this situation, the overall system combination of multiple systems is effective; however, the computational resources increase in proportion to the number of systems. This paper proposes a method to select an optimal single system from multiple systems. The selection is based on the estimated word error rates of a respective system by using the i-vector similarities between training and test data. The experiments on the third CHiME challenge show that our proposed method can efficiently select a single system from multiple systems with different speech enhancement and feature transformation methods to improve the overall performance without increasing computational resources.**

I. Introduction

Automatic speech recognition (ASR) is a key component for many "hands-free" applications in various environments. Speech applications are widely used, because speech input is faster than keyboard input. To increase the practicality of ASR systems, distant-talking input is much more desirable than close-talking input; however, noise or interferences significantly degrade ASR performance. To address this problem, many methods have been proposed to improve ASR performance under noisy environments such as speech enhancement (SE), feature transformation, and discriminative methods. Each method has a specialty for specific noise and no universal solution exists. The optimal ASR system is different for each utterance and the number of their combination is enormous. Although overall combination of their hypotheses can improve the performance, the computational resources increase in proportion to the number of systems. If the single optimal system can be picked up from many ASR systems prior to SE and ASR decoding, the computational resources do not increase. For example, if there are two systems and the first system is apparently superior to the second system for environment A and the second system is superior to the first system for environment B, it is better to select an optimal single system than to combine systems in terms of a computational resource.

In this paper, we propose an efficient system selection method based on the estimated word error rates (WERs) of ASR systems before performing SE and ASR decoding. Previous studies [1], [2] used perceptual evaluation speech quality (PESQ) scores to predict WERs; however, the calculation of PESQ scores requires clean speech, which cannot be obtained for evaluation data. Even if PESQ scores can be obtained, these kinds of estimations also require enhanced speech, thus it is necessary to perform at least SE before selection. On the other hand, limited to reverberation, the performance is estimated from room acoustic parameters [2], [3] but these types of estimations need room acoustic impulse responses and this is not a realistic assumption. Another study [4] used recognition hypotheses; however, in order to select the optimal SE method, it is inefficient to perform SE and ASR decoding for every system. In addition, if multiple hypotheses have been already obtained, system combination is better than system selection. Our method uses i-vectors, which represent speaker and channel characteristics [5], [6] of original noisy speech for estimating WERs via cosine similarities between the training and test data. It is unnecessary to perform not only ASR decoding but also SE. A related approach is [7], which uses i-vectors for clustering training data but whose objective is different from our approach.

This paper validates the effectiveness of the proposed approach on the third Computational Hearing in Multisource Environments (CHiME) challenge [8]. The third one has been released after the success of two challenges: the first CHiME challenge, which was a simple keyword recognition task [9] and the second CHiME challenge, which additionally contained a medium vocabulary recognition task (track 2) [10]. We showed the effectiveness of SE and various state-of-the-art ASR techniques for this second CHiME challenge track 2 [11], [12]. The third CHiME challenge is also a medium vocabulary task, which aims to improve the performance of ASR systems in four different public environments such as cafés or streets by using six tablet-embedded microphones. In addition, there are two different conditions in the third CHiME challenge: real ("Real") and simulation data ("Sim."). To overcome this challenging task, we prepare multiple ASR systems with different SE methods and various feature transformations. As mentioned above, the optimal system is different for each environment. In this case, the SE method attached to the challenge baseline performs well for Sim., whereas our employed SE method (maximum signal-to-noise ratio (SNR) beamformer (BF) [13]) performs well for Real. For this type of situation, system combinations–e.g., recognizer output voting error reduction (ROVER) [14]–can refine hypotheses by majority voting of the hypotheses of multiple systems. Actually, when increased computational resources can be ignored, system combination is a more robust solution for mismatch and diversity of

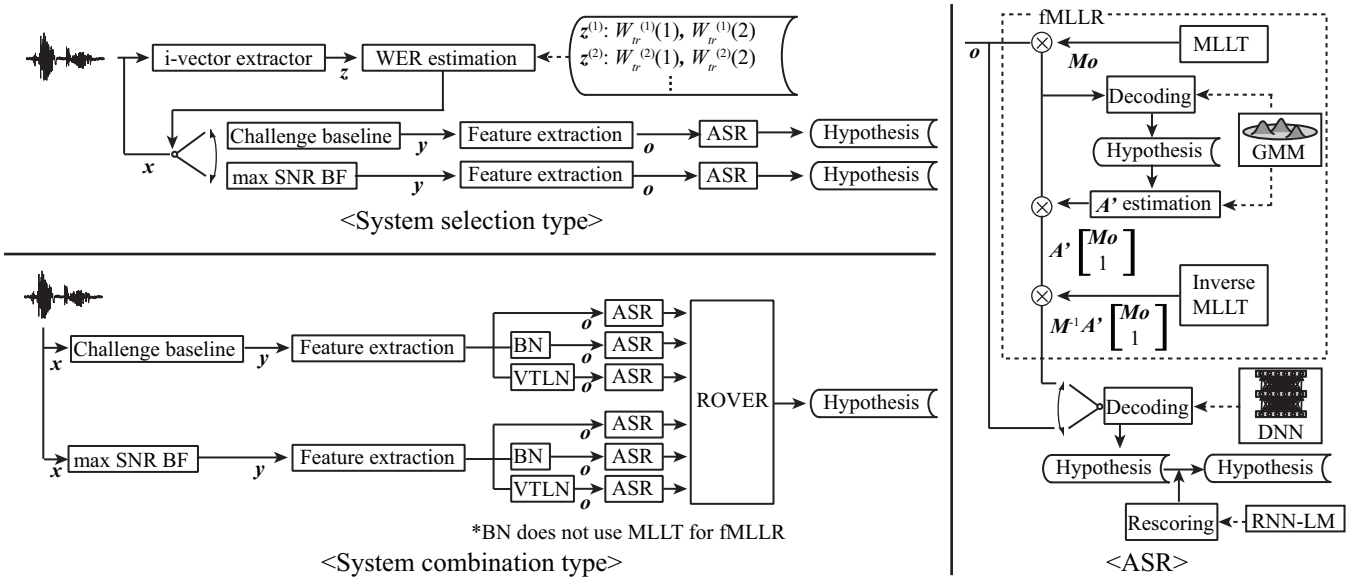

Fig. 1. Schematic diagram of the proposed ASR systems.

environments [15], [16]. Experiments show that the proposed optimal ASR system selection method is effective to exploit the better performance from multiple different systems without increasing computational resources.

II. System overview

Figure 1 shows two types of systems; one is the proposed system selection type and the other is a conventional system combination type. The system selection type selects a single system based on i-vectors (Section V) whereas the system combination type combines multiple systems' hypotheses to refine the hypotheses by ROVER. There are multiple systems using different SE methods and different feature transformations. Each system has a noise suppression component (CHiME challenge-provided baseline and max SNR BF (Section III)) and an ASR decoding component. The ASR decoding component uses either Gaussian mixture model (GMM) or deep neural network (DNN) acoustic model with sequence discriminative training (Section IV-C) after feature transformation including bottleneck (BN) features and vocal tract length normalizations (VTLNs) and feature adaptation (Section IV-A and IV-B). In addition, rescoring of language model scores is used by an interpolation of original tri-gram model scores and recurrent neural network language model (RNN-LM) scores.

III. Speech enhancement methods

SE is performed before ASR and a blind SE method is used because speaker positions are unstable. Two types of blind methods are prepared.

A. Challenge baseline

This method estimates a direction of arrival by a nonlinear SRP-PHAT pseudo-spectrum [17]. After target direction is obtained, Viterbi algorithm is used for calculating transition probabilities between successive speaker positions. These probabilities are related to the distance between the speaker and microphone array. The multichannel spatial covariance matrices are estimated from noise signals in 5 seconds, which are added before the speech. Using these matrices, time-varying minimum variance distortionless response beamforming with diagonal loading [18] enhances speech with taking possible microphone failures into account.

B. Maximum SNR BF

In addition to the challenge baseline, we employ a maximum signal-to-noise ratio (max SNR) beamformer (BF) [13], which is one of the statistically optimal BFs [19]. The enhanced speech spectrum at frame t and frequency bin ω, $y_{t,\omega} \in \mathbb{C}$, is obtained from $N_c$ch original spectrum $\boldsymbol{x}_{t,\omega} \in \mathbb{C}^{N_c \times 1}$ with a mask $\boldsymbol{w}_\omega \in \mathbb{C}^{1 \times N_c}$:

$$y_{t,\omega} = \boldsymbol{w}_\omega \boldsymbol{x}_{t,\omega}. \tag{1}$$

According to the voice activity detection results, SNR λ_ω is defined as

$$\lambda_\omega = \frac{\boldsymbol{w}_\omega \boldsymbol{R}_s \boldsymbol{w}_\omega^{\mathrm{H}}}{\boldsymbol{w}_\omega \boldsymbol{R}_n \boldsymbol{w}_\omega^{\mathrm{H}}}, \tag{2}$$

where $\boldsymbol{R}_s$ and $\boldsymbol{R}_n$ are covariance matrices in the speech and noise frames, respectively, and H denotes the Hermitian transpose operation. The mask $\boldsymbol{w}_\omega$ that maximizes SNR λ_ω corresponds to a solution to a general eigenvalue problem:

$$\boldsymbol{w}_\omega \boldsymbol{R}_s^{\mathrm{H}} = \lambda_\omega \boldsymbol{w}_\omega \boldsymbol{R}_n^{\mathrm{H}}. \tag{3}$$

IV. DNN with feature transformation and discriminative training

A. Feature-space adaptation for DNN

To normalize the large variations of features among speakers and noises, feature adaptation is still effective for DNN. This paper validates feature-space maximum likelihood linear regression (fMLLR) [20] with speaker adaptive training (SAT) [21]. Feature adaptation methods can improve ASR accuracies in noisy environments by adapting to unknown and

changing noise conditions [20], [22]. Conventional fMLLR is applied for DNN [23], [24], [25] after ASR decoding is performed using GMM. fMLLR types of feature adaptations maximize a likelihood $\mathcal{L}$ for normal distributions $\mathcal{N}$ of the j-th state and m-th mixture with the mean $\boldsymbol{\mu}_{jm}$ and covariance $\boldsymbol{\Sigma}_{jm}$ as

$$\mathcal{L}_{jm}(\boldsymbol{o}_t) = |\boldsymbol{A}|\,\mathcal{N}\left(\hat{\boldsymbol{o}}_t|\boldsymbol{\mu}_{jm}, \boldsymbol{\Sigma}_{jm}\right), \tag{4}$$

where $\boldsymbol{o}_t$ is an observation at frame t and $\hat{\boldsymbol{o}}_t$ is a transformed feature as

$$\hat{\boldsymbol{o}}_t \triangleq \boldsymbol{A}\boldsymbol{o}_t + \boldsymbol{b} = \boldsymbol{A}' \begin{bmatrix} \boldsymbol{o}_t \\ 1 \end{bmatrix}. \tag{5}$$

After adaptation, transformed features $\hat{\boldsymbol{o}}_t$ can be input in the same manner as the original feature. However, widely-used filter bank (fbank) features cannot be represented well by a diagonal covariance GMM [23]. For this limitation, fMLLR with fbank features did not improve the ASR performance [26] and it is necessary to de-correlate fbank features before adaptation. In the adaptation phase, a global maximum likelihood linear transformation (MLLT) [27] $\boldsymbol{M}$ is applied to de-correlate fbank features, whereas in the decoding phase, an inverse MLLT $\boldsymbol{M}^{-1}$ is applied to de-correlated and adapted fMLLR features as

$$\hat{\boldsymbol{o}}_t = \boldsymbol{M}^{-1}\boldsymbol{A}' \begin{bmatrix} \boldsymbol{M}\boldsymbol{o}_t \\ 1 \end{bmatrix}. \tag{6}$$

B. BN and VTLN features

Two types of additional feature transformations are investigated: BN features [28], [29] and VTLN [30], [31], [32]. Before DNN prevailed, to combine neural networks with a conventional GMM, a tandem structure was used [33]. This approach has been extended to DNN and its extension–the BN feature–is widely used because conventional GMM can be used for decoding and features can be easily adapted for these types of structures. The BN feature is a lower dimensional hidden-layer unit output. To extract BN features, DNN is trained to predict phoneme states when the hidden layer size is smaller than the input layer size.

VTLN is another type of speaker normalization technique. Among several VTLN methods, we employ a simple linear approximation approach [32]. To approximate usual VTLN warped features $\boldsymbol{o}_t^\alpha$ with different warping factors α's, linear VTLN uses linear transformations $\boldsymbol{A}^\alpha$ and offsets $\boldsymbol{b}^\alpha$, which map an original feature $\boldsymbol{o}_t$ to the warped feature $\boldsymbol{o}_t^\alpha$ as

$$\boldsymbol{o}_t^\alpha \approx \boldsymbol{A}^\alpha \boldsymbol{o}_t + \boldsymbol{b}^\alpha. \tag{7}$$

These parameters ($\boldsymbol{A}^\alpha$ and $\boldsymbol{b}^\alpha$) are obtained to minimize square errors

$$\boldsymbol{A}^\alpha, \boldsymbol{b}^\alpha \leftarrow \underset{\boldsymbol{A}^\alpha, \boldsymbol{b}^\alpha}{\arg\min} \left|\boldsymbol{o}_t^\alpha - (\boldsymbol{A}^\alpha \boldsymbol{o}_t + \boldsymbol{b}^\alpha)\right|^2, \tag{8}$$

by using a subset of training data.

C. sMBR discriminative training of DNNs

Since ASR is a sequence-level pattern recognition problem, the performance of the sequence-level pattern recognition is more important than that of the frame-level. DNNs are already discriminative at the frame level cross entropy, however, sequence-level discriminative training further minimizes the classification errors on the whole sequence [34]. Thus, the hybrid architecture with hidden Markov models (HMMs) has still been a mainstream for ASR.

A DNN model with parameters θ outputs posterior probabilities $p_\theta(j|\boldsymbol{o}_t)$ of the j-th HMM state. These probabilities are computed using a softmax layer:

$$p_\theta(j|\boldsymbol{o}_t) = \frac{\exp a_\theta(j|\boldsymbol{o}_t)}{\sum_{j'} \exp a_\theta(j'|\boldsymbol{o}_t)}, \tag{9}$$

where a_θ is the output of the top layer. Each layer of the DNN transforms the outputs of the previous layer through an affine transform, whose parameters are a subset of θ, followed by a non-linear operation such as a sigmoid.

In order to use the classical HMM-based decoding framework, hybrid DNN-HMM systems replace the acoustic likelihood of GMMs with a pseudo-likelihood $p_\theta\left(\boldsymbol{o}_t|j\right)$ as

$$p_\theta\left(\boldsymbol{o}_t|j\right) \propto \frac{p_\theta\left(j|\boldsymbol{o}_t\right)}{p_0\left(j\right)}, \tag{10}$$

where $p_0\left(j\right)$ is the prior probability calculated from the count of the states in the training data.

The parameters θ are trained discriminatively according to the sequence-level minimum Bayes risk (sMBR) criterion:

$$\mathcal{F}_{\mathrm{sMBR}}(\theta) = \sum_r \frac{\sum_s p_\theta\left(\boldsymbol{o}^{(r)}|\mathcal{H}_s\right)^\kappa p_L(s) A(s, s^{(r)})}{\sum_s p_\theta\left(\boldsymbol{o}^{(r)}|\mathcal{H}_s\right)^\kappa p_L(s)}, \tag{11}$$

where $\boldsymbol{o}^{(r)}$ is the rth utterance observation vector $(\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots)$; s is a hypothesis of the ASR systems for the reference $s^{(r)}$; p_L is the likelihood of a language model; κ is an acoustic scale; and A is the raw frame accuracy. The gradient of the objective function with respect to a_θ can be obtained as

$$\begin{aligned} \frac{\partial \mathcal{F}_{\mathrm{sMBR}}(\theta)}{\partial a_\theta(j)} &= \sum_{j'} \frac{\partial \mathcal{F}_{\mathrm{sMBR}}(\theta)}{\partial \log p_\theta\left(\boldsymbol{o}^{(r)}|j'\right)} \frac{\partial \log p_\theta\left(\boldsymbol{o}^{(r)}|j'\right)}{\partial a_\theta(j)}, \\ &= \kappa \gamma_{j,t} \left(\hat{A}(j) - \hat{A}\right), \end{aligned} \tag{12}$$

where $\hat{A}(j)$ is the average accuracy of all hypotheses in the lattice whose state at frame t is j; $\hat{A}$ is the average accuracy of all hypotheses; and $\gamma_{j,t}$ is the posteriors of state j for all hypotheses in the lattice. The back-propagation procedure with Eq. (12) updates θ.

V. Optimal ASR system selection based on an estimated WER via i-vector similarities

We propose an efficient optimal system selection method that estimates the best performing single system among multiple systems for an unknown utterance based on the i-vector [5], [6]. For all training data, WERs per utterance, W_{tr}, are obtained a priori.

Algorithm 1 Algorithm of the proposed optimal system selection method

Input: i-vector for all training data $\boldsymbol{z}_{tr}$, and WER for all training data and all prepared ASR systems $W_{tr}(i)$ where i is a system ID

for $r_{ev} = 1$ to (# of evaluation utterances) **do**
  Extract i-vector $\boldsymbol{z}_{ev}^{(r_{ev})}$
  for $r_{tr} = 1$ to (# of training utterances) **do**
    Compute similarities $\sigma\left(\boldsymbol{z}_{ev}^{(r_{ev})}, \boldsymbol{z}_{tr}^{(r_{tr})}\right)$
  end for
  Find the most similar utterance $\hat{r}_{tr}$ as in Eq. (14)
  Find the best ASR system $\hat{i}$ for the utterance $\hat{r}_{tr}$ as in Eq. (16)
end for

Output: The optimal system IDs for all evaluation utterances

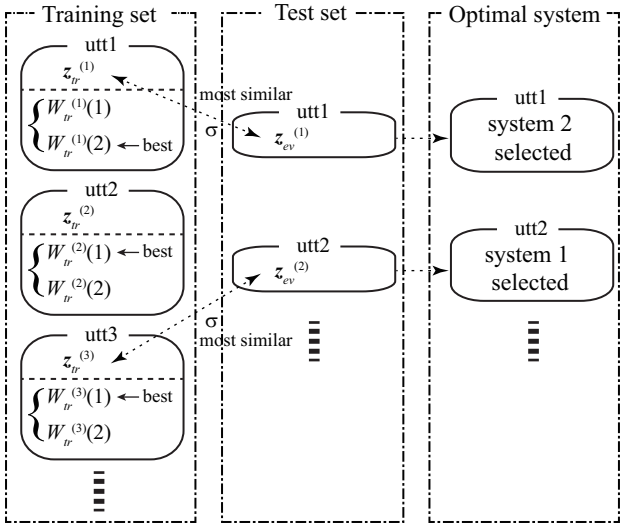

Fig. 2. Example of the proposed optimal system selection method.

i-vectors are derived from a factor analysis that decomposes speech into a speaker/channel invariant part and a variant part as

$$\boldsymbol{V}^{(r)} = \boldsymbol{v} + \boldsymbol{T}\boldsymbol{z}^{(r)}, \tag{13}$$

where $\boldsymbol{V}^{(r)}$ is a GMM super vector adapted to the utterance r and is dependent on a speaker and a channel; $\boldsymbol{v}$ is a GMM super vector, which is independent of the speaker and the channel and is obtained from a universal background model; $\boldsymbol{T}$ is a low-rank rectangular matrix composed of basis vectors that span all variable spaces; and $\boldsymbol{z}^{(r)}$ is an i-vector for an utterance r.

Utterance similarities σ are calculated from i-vectors for evaluation data $\boldsymbol{z}_{ev}^{(r_{ev})}$ and those for training data $\boldsymbol{z}_{tr}^{(r_{tr})}$. The most similar utterance $\hat{r}_{tr}$ to the evaluation data r_{ev} is picked up from the training data as

$$\hat{r}_{tr} \leftarrow \arg\max_{r_{tr}} \sigma\left(\boldsymbol{z}_{ev}^{(r_{ev})}, \boldsymbol{z}_{tr}^{(r_{tr})}\right). \tag{14}$$

For similarity, e.g., cosine similarity (15) can be used.

$$\sigma\left(\boldsymbol{z}_{ev}^{(r_{ev})}, \boldsymbol{z}_{tr}^{(r_{tr})}\right) = \frac{\boldsymbol{z}_{ev}^{(r_{ev})} \cdot \boldsymbol{z}_{tr}^{(r_{tr})}}{\left|\boldsymbol{z}_{ev}^{(r_{ev})}\right| \left|\boldsymbol{z}_{tr}^{(r_{tr})}\right|}. \tag{15}$$

After the most similar utterance is found in the training data, the optimal system $\hat{i}$ is selected in the reference of WERs of training data as in Eq. (16) because similar utterances ought to have similar ASR performances.

$$\hat{i} \leftarrow \arg\min_{i} W_{tr}^{(\hat{r}_{tr})}(i). \tag{16}$$

Here, $W_{tr}^{(\hat{r}_{tr})}(i)$ is a WER of the i-th system for the utterance $\hat{r}_{tr}$[1]. Algorithm 1 shows the detailed procedure of the proposed method.

Fig. 2 shows an example of the proposed optimal system selection. In this case, there are two systems. First, respective WERs for all training data utterances are obtained. Next, for the given test data, i-vectors are calculated and the most similar utterance in the training data is found based on the i-vector similarity σ. In this case, the most similar utterance of the first utterance of the test data is the first utterance in the training data. Finally, in the reference of WER of the most similar utterance, the optimal system is selected. For the first utterance of the test data, the system two is selected because the WER of the second system is better than that of the first system.

VI. Experimental setups

We validated the effectiveness of our proposed approach for the third CHiME challenge [8]. As mentioned in the introduction, this is a medium-vocabulary noisy ASR task whose speech utterances are taken from the *Wall Street Journal* database. There are two types of data: real data ("Real") and simulated data ("Sim."). The real data were recorded in the real world, whereas the simulated data were created by convolving

TABLE I
Number of utterances and speakers in each dataset of the third CHiME challenge.

	# utterances		# speakers	
dataset	Real	Sim.	Real	Sim.
Training set	1,600	7,138	4	83
Development set	1,640	1,640	4	4
Evaluation set	1,320	1,320	4	4

TABLE II
Setup for the ASR systems.

Sampling frequency	16 kHz
Window length	25 ms
Window shift	10 ms
Features (GMM)	0–12th MFCCs + Δ + ΔΔ
Features (DNN)	0–22th filter banks + Δ + ΔΔ
HMM states	2,500 shared triphone states
Number of Gaussians	15,000
DNN nodes per layer	1024 nodes
DNN layer size	7 layers
Vocabulary size	5,000

[1]An average or an weighted average of WERs of the N-best results can be used for W_{tr} instead of WERs of the 1-best results.

clean speech with impulse responses and adding noise. Each type of data has four environments: bus, café, pedestrian, and street. WERs below are averaged over four environments. Table I shows the dataset description. The training set has 1,600 and 7,138 utterances by 4 and 83 speakers for Real and Sim, respectively. The development (Dev.) and evaluation (Eval.) set have 1,640 and 1,320 utterances, respectively, by 4 speakers both for Real and Sim. This paper evaluated noisy speech, challenge-provided enhanced speech ("enh1"), and our enhanced speech ("enh2"). After multiple systems were constructed with two types of SE and various feature transformations, the optimal systems were selected by our proposed method or their hypotheses were combined by ROVER. Finally, language model scores were rescored by interpolating n-gram language model scores and recurrent neural network language model (RNN-LM) scores [35], [36]. The setups for RNN-LM were the same to those attached to the Kaldi WSJ example.

There were two types of acoustic feature settings. The first setting was MFCC with feature transformations. In addition to the standard 0–12th order MFCC features with Δ and $\Delta\Delta$, linear discriminant analysis (LDA) [37] compressed the static MFCCs in nine contiguous frames into 40-dimensional features before a global MLLT [27] was applied. The second setting started from the 0–22nd order fbank features with Δ and $\Delta\Delta$. For fMLLR, MLLT was used to de-correlate the features before adaptation. For both settings, to reduce the variances between speakers, SAT [21] was used where training is conducted after having transformed the training speech into a canonical space. The BN feature was a 40-dimensional hidden-layer unit output of DNN with two hidden layers. The warping parameters of linear VTLN were changed from 0.85 to 1.25 with a step of 0.01.

We trained DNNs after GMMs by using the Kaldi toolkit [38]. Table II shows the ASR setup. The detailed training procedure of GMMs was in [11], [12]. The number of monophones was 40, including silence. The number of context-dependent tri-phone states was 2,500 and the total number of Gaussians was 15,000. The parameters used in our experiments were the same to those in the challenge provided baseline. We used "nnet1" of the Kaldi toolkit for DNN training. Starting from the seven-layer restricted Boltzmann machine, the DNN was constructed where each hidden layer has a sigmoid activation. The learning rate was decreased from the initial learning rate (0.008) if the decrease of CE in the development set was under the threshold. Features across nine concatenated frames were inputted and the number of nodes per hidden layer was 1,024. We investigated the performance change when using feature-space boosted maximum mutual information (f-bMMI) [39] for GMM and sMBR for DNN.

VII. Results and discussions

A. GMM-based baseline ASR systems

Table III shows the average WER of GMM-based ASR systems on the Dev. and Eval. set. For all cases, SE improved the performance; "enh1" significantly improved the WER for Sim. but provided little improvement for Real. On the other hand, "enh2" significantly improved the WER for Real but was less effective for Sim. than "enh1". Feature transformation and adaptation (FTA in the figure) led to the WER improvement of 7–11%. From now on, the WER improvements were evaluated in terms of an absolute value. Discriminative training (DT in the figure) resulted in the additional WER improvements of approximately 2–3%. Even after SE, these techniques were still effective. These tendencies were similar to those of the second CHiME challenge [11], [12].

TABLE III
Average WER [%] on the development and evaluation set of the third CHiME challenge using GMM acoustic models. The effectiveness of feature transformation and adaptation (FTA) and discriminative training (DT) is shown. Two types of SE methods (enh1 (challenge baseline) and enh2 (max SNR BF)) were evaluated in addition to noisy speech.

	FTA	DT	Dev. set		Eval. set	
			Real	Sim.	Real	Sim.
noisy			26.90	24.40	43.06	30.70
	✓		18.44	17.74	31.87	21.96
		✓	16.04	14.78	27.05	17.16
enh1			26.80	13.51	47.66	15.65
	✓		19.92	9.76	35.78	11.16
		✓	17.70	**7.60**	32.12	**8.97**
enh2			21.35	16.51	36.49	22.77
	✓		14.76	11.70	27.41	16.25
		✓	**12.43**	9.05	**21.61**	13.33

B. DNN-based ASR systems

Table IV shows the average WER of DNN-based ASR systems. The tendencies were similar to those in GMM-based systems (VII-A). sMBR of DNN improved the WER by 1–2% especially effective for "enh2". fMLLR based model adaptation improved the WER by 1–3% but SAT was less effective (less than 1%). The BN feature was effective for Sim. but ineffective for Real. The VTLN provided an additional improvement on the Dev. set but worsened the WERs on Sim. of the Eval. set. These ASR systems were combined (section VII-C) or selected (section VII-D) because their performance tendencies were different from environment to environment.

C. ASR system combination

Table V (C) shows the results of two, three, or six system combinations. Increasing the number of systems did not necessarily lead to the performance improvement because the best performing systems were different as shown in Table IV. For Dev. set, certainly, six system combination was the best for Sim. but for Real, three system combination was the best. For the reference, table also shows the WER of the best (B in the table) or the worst single system (W) from six systems. All systems were better than the worst system and some systems outperformed the best single system. This shows the effectiveness of system combination in exchange for the increase of computational resources. Rescoring with RNN-LM improved the WER further by 1–2%. Considering longer context than n-gram model was effective.

TABLE IV
AVERAGE WER [%] ON THE DEVELOPMENT AND EVALUATION SET OF THE THIRD CHIME CHALLENGE USING DNN ACOUSTIC MODELS.

	BN	VTLN	fMLLR	SAT	sMBR	RNN-LM	Dev. set		Eval. set		
							Real	Sim.	Real	Sim.	
noisy							15.58	13.51	29.21	18.41	
					✓		14.41	12.62	28.49	16.90	
			✓		✓		12.11	11.40	22.74	13.57	
			✓	✓	✓		12.05	11.16	22.25	13.95	
	✓		✓	✓	✓		12.31	10.75	22.91	12.67	
enh1							17.64	7.44	32.03	9.04	
					✓		16.51	7.01	30.84	8.26	
			✓	✓	✓		13.65	6.04	24.32	7.04	1-a
	✓		✓	✓	✓		14.42	5.92	26.17	6.33	1-b
		✓	✓	✓	✓		13.11	5.90	20.22	12.45	1-c
			✓	✓	✓	✓	11.88	4.65	21.66	5.22	
	✓		✓	✓	✓	✓	12.78	**4.41**	24.11	**4.75**	
		✓	✓	✓	✓	✓	11.36	4.57	17.93	10.23	
enh2							12.83	9.38	25.94	14.57	
					✓		11.36	8.39	22.41	12.84	
			✓	✓	✓		9.03	7.08	16.98	10.45	2-a
	✓		✓	✓	✓		9.67	6.89	17.74	9.99	2-b
		✓	✓	✓	✓		13.97	6.11	20.02	13.69	2-c
			✓	✓	✓	✓	**7.39**	5.69	**14.79**	8.65	
	✓		✓	✓	✓	✓	8.02	5.48	15.59	8.19	
		✓	✓	✓	✓	✓	12.18	4.76	17.64	12.08	

TABLE V
AVERAGE WER [%] ON THE DEVELOPMENT AND EVALUATION SET USING SYSTEM SELECTION (S) AND SYSTEM COMBINATION (C). FOR REFERENCE, THE BEST SYSTEM (B) AND THE WORST SYSTEM (W) WERE PICKED UP. ADDITIONALLY, RESCORING WITH RNN-LM WAS PERFORMED.

Type	# of systems	Target systems						RNN-LM	Dev. set		Eval. set	
		1-a	1-b	1-c	2-a	2-b	2-c		Real	Sim.	Real	Sim.
B	1 from 6	✓	✓	✓	✓	✓	✓		9.03	5.90	16.98	6.33
W	1 from 6	✓	✓	✓	✓	✓	✓		14.42	7.08	26.17	13.69
C	2	✓			✓				8.71	5.86	16.38	7.08
C	3	✓	✓	✓					12.73	5.51	19.59	**6.28**
C	3				✓	✓	✓		**8.23**	5.73	16.31	9.78
C	6	✓	✓	✓	✓	✓	✓		10.02	**5.27**	**15.67**	7.42
S	1 from 2	✓			✓				*10.10*	6.81	19.72	9.89
S	1 from 3	✓	✓	✓					14.24	*5.98*	25.67	*7.35*
S	1 from 3				✓	✓	✓		10.45	6.70	*18.52*	10.60
S	1 from 6	✓	✓	✓	✓	✓	✓		11.36	6.52	20.60	9.28
B	1 from 6	✓	✓	✓	✓	✓	✓	✓	7.39	4.41	14.79	4.75
W	1 from 6	✓	✓	✓	✓	✓	✓	✓	12.78	5.69	24.11	12.08
C	2	✓			✓			✓	7.52	4.59	14.61	5.72
C	3	✓	✓	✓				✓	11.09	4.15	17.61	**4.82**
C	3				✓	✓	✓	✓	**6.66**	4.54	14.15	8.39
C	6	✓	✓	✓	✓	✓	✓	✓	8.70	**3.93**	**13.74**	5.97
S	1 from 2	✓			✓			✓	*8.49*	5.39	17.45	8.10
S	1 from 3	✓	✓	✓				✓	12.49	*4.55*	23.06	*5.49*
S	1 from 3				✓	✓	✓	✓	8.64	5.28	*16.41*	8.83
S	1 from 6	✓	✓	✓	✓	✓	✓	✓	9.57	5.14	18.49	7.92

D. Optimal ASR system selection

Our proposed method based on i-vectors selected the optimal single system from a combination of two types of SE methods and three types of feature transformations. Table V (S) shows the results. For Real, "enh1" tended to be picked up and for Sim. "enh2" tended to be picked up. All system selections were better than the worst system. This shows the effectiveness of the proposed method, because the proposed method aims to pick up the best system. The average differences between the best system –upper limit of a single system ASR– and the proposed system were 0.58% for Dev. set and 1.28% for Eval. set. In total, the worst WER of the selected system for either Real or Sim. was better than that of each single system. Tendencies were the same to the case of rescoring with RNN-LM. The average differences between the best system and the proposed system were 0.62% for Dev. set and 1.18% for Eval. set. The performance differences were larger for Eval. set than for Dev. set because Eval. set had larger mismatches between training and test data and the performance was worse.

VIII. CONCLUSION AND FUTURE WORK

This paper proposed an efficient optimal system selection method that estimates WERs of a test utterance based on the i-vector similarities when there are multiple ASR systems and their suitable environments are different. The proposed system selection can improve the worst performance for single

systems by picking up better hypotheses. The experiments on the third CHiME challenge showed that the average differences between the best WER of the single system and that of the selected system were around 0.6% for the development set and 0.9% for the evaluation set. This shows the effectiveness of our proposed method. Our method does not increase the computational resources, although system combination improved the performance further but it increases the computational resources in proportion to the number of combined ASR systems. Future work will be a precise estimation of WER by using data clustering or an average of WERs of the N-best results.

REFERENCES

[1] T. Yamada, M. Kumakura, and N. Kitawaki, "Performance estimation of speech recognition system under noise conditions using objective quality measures and artificial voice," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, pp. 2006–2013, 2006.

[2] T. Fukumori, M. Nakayama, T. Nishiura, and Y. Yamashita, "Estimation of speech recognition performance in noisy and reverberant environments using PESQ score and acoustic parameters," in *Proceedings of APSIPA ASC*, 2013.

[3] A. Brutti and M. Matassoni, "On the use of early-to-late reverberation ratio for ASR in reverberant environments," in *Proceedings of ICASSP*, 2014, pp. 4638–4642.

[4] A. Ogawa and A. Nakamura, "Joint estimation of confidence and error causes in speech recognition," *Speech Communication*, vol. 54, pp. 1014–1028, 2012.

[5] N. Dehak, *Discriminative and generative approaches for long- and short-term speaker characteristics modeling: Application to speaker verification*, Ph.D. dissertation, cole de Technologies Suprieure, 2009.

[6] N. Dehak, P.J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 5 2011.

[7] O. Siohan and M. Bacchiani, "ivector-based acoustic data selection," in *Proceedings of INTERSPEECH*, 2013.

[8] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'CHiME' speech separation and recognition challenge: Dataset, task and baselines," in *Proceedings of ASRU*, 2015.

[9] J. Barker, E. Vincent, N. Ma, C. Christensen, and P. Green, "The PASCAL CHiME speech separation and recognition challenge," *Computer Speech and Language*, vol. 27, no. 3, pp. 621–633, 2013.

[10] E. Vincent, J. Barker, S. Watanabe, J. Le Roux, F. Nesta, and M. Matassoni, "The second 'CHiME' speech separation and recognition challenge: Datasets, tasks and baselines," in *Proceedings of ICASSP*, 2013, pp. 126–130.

[11] Y. Tachioka, S. Watanabe, J. Le Roux, and J.R. Hershey, "Discriminative methods for noise robust speech recognition: A CHiME challenge benchmark," in *Proceedings of the 2nd CHiME Workshop on Machine Listening in Multisource Environments*, 2013, pp. 19–24.

[12] Y. Tachioka, S. Watanabe, and J.R. Hershey, "Effectiveness of discriminative training and feature transformation for reverberated and noisy speech," in *Proceedings of ICASSP*, 2013, pp. 6935–6939.

[13] S. Araki, H. Sawada, and S. Makino, "Blind speech separation in a meeting situation," in *Prodeedings of ICASSP*, 2007, vol. 1, pp. 41–45.

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

[15] Y. Tachioka, T. Narita, S. Watanabe, and F. Weninger, "Dual system combination approach for various reverberant environments," in *Proceedings of REVERB challenge*, 2014, pp. 1–8.

[16] H. Hermansky, L. Burget, J. Cohen, E. Dupoux, N. Feldman, J. Godfrey, S. Khudanpur, M. Maciejewski, S.H. Mallidi, A. Menon, T. Ogawa, V. Peddinti, R. Rose, R. Stern, M. Wiesner, and K. Vesely, "Towards machines that know when they do not know: Summary of work done at 2014 Frederick Jelinek memorial workshop," in *Proceedings of ICASSP*, 2015, pp. 5009–5013.

[17] B. Loesch and B. Yang, "Adaptive segmentation and separation of determined convolutive mixtures under dynamic conditions," in *Proceedings of the 9th International Conference on Latent Variable Analysis and Signal Separation (LVA/ICA)*, 2010, pp. 41–48.

[18] X. Mestre and M.A. Lagunas, "On diagonal loading for minimum variance beamformers," in *Proceedings of the IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 2003, pp. 459–462.

[19] R. Monzingo and T. Miller, *Introduction to adaptive arrays*, Wiley and Sons, 1980.

[20] M.J.F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[21] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Proceedings of ICSLP*, 1996, pp. 1137–1140.

[22] K. Shinoda and C.H. Lee, "A structural Bayes approach to speaker adaptation," *IEEE Transactions on Speech and Audio Processing*, vol. 9, pp. 276–287, 2001.
[23] T.N. Sainath, B. Kingsbury, A. Mohamed, G.E. Dahl, G. Saon, H. Soltau, T. Beran, A.Y. Aravkin, and B. Ramabhadran, "Improvements to deep convolutional neural networks for LVCSR," in *Proceedings of ASRU*, 2013, pp. 315–320.
[24] T. Yoshioka, A. Ragni, and M.J.F. Gales, "Investigation of unsupervised adaptation of DNN acoustic models with filter bank input," in *Proceedings of ICASSP*, 2014, pp. 13–16.
[25] H. Kanagawa, Y. Tachioka, S. Watanabe, and J. Ishii, "Feature-space structural MAPLR with regression tree-based multiple transformation matrices for DNN," in *Proceedings of APSIPA*, 2015.
[26] T.N. Sainath, A. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for LVCSR," in *Proceedings of ICASSP*, 2013, pp. 8614–8618.
[27] R.A. Gopinath, "Maximum likelihood modeling with Gaussian distributions for classification," in *Proceedings of ICASSP*, 1998, pp. 661–664.
[28] F. Grézl, M. Karafiát, S. Kontár, and J. Černocký, "Probabilistic and bottle-neck features for LVCSR of meetings," in *Proceedings of ICASSP*, 2007, vol. 4, pp. 757–760.
[29] D. Yu and M.L. Seltzer, "Improved bottleneck features using pretrained deep neural networks," in *Proceedings of INTERSPEECH*, 2011, pp. 237–240.
[30] E. Eide and H. Gish, "A parametric approach to vocal tract length normalization," in *Proceedings of ICASSP*, 1996, vol. 1, pp. 346–3483.
[31] S. Umesh, A. Zolnay, and H. Ney, "Implementing frequency-warping and VTLN through linear transformation of conventional MFCC," in *Prodeedings of INTERSPEECH*, 2005, pp. 269–272.
[32] S. Panchapagesan and A. Alwan, "Frequency warping for VTLN and speaker adaptation by linear transformation of standard MFCC," *Computer Speech and Language*, vol. 23, no. 1, pp. 42–64, 1 2009.
[33] H. Hermansky, D.P.W. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional HMM systems," in *Proceedings of ICASSP*, 2000.
[34] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proceedings of INTERSPEECH*, 2013.
[35] T. Mikolov, M. Karafiát, L. Burget, J. Černocký, and S. Khudanpur, "Recurrent neural network based language model," in *Proceedings of INTERSPEECH*, 2010, pp. 1045–1048.
[36] Y. Tachioka and S. Watanabe, "Discriminative method for recurrent neural network language models," in *Proceedings of ICASSP*, 2015, pp. 5386–5390.
[37] R. Haeb-Umbach and H. Ney, "Linear discriminant analysis for improved large vocabulary continuous speech recognition," in *Proceedings of ICASSP*, 1992, pp. 13–16.
[38] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, M. Petr, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Veselý, "The Kaldi speech recognition toolkit," in *Proceedings of ASRU*, 2011, pp. 1–4.
[39] D. Povey, B. Kingsbury, L. Mangu, G. Saon, H. Soltau, and G. Zweig, "fMPE: Discriminatively trained features for speech recognition," in *Proceedings of ICASSP*, 2005, pp. 961–964.