Interspeech 2018
2-6 September 2018, Hyderabad, India

Auxiliary feature based adaptation of end-to-end ASR systems

Marc Delcroix[1], *Shinji Watanabe*[2], *Atsunori Ogawa*[1], *Shigeki Karita*[1], *Tomohiro Nakatani*[1]

[1]NTT Communication Science Laboratories, NTT Corporation, Kyoto, Japan
[2]Center for Language and Speech Processing, Johns Hopkins University, Baltimore, MD, USA
{marc.delcroix,ogawa.atsunori,karita.shigeki,nakatani.tomohiro}@lab.ntt.co.jp,
shinjiw@jhu.edu

Abstract

Acoustic model adaptation has been widely used to adapt models to speakers or environments. For example, appending auxiliary features representing speakers such as i-vectors to the input of a deep neural network (DNN) is an effective way to realize unsupervised adaptation of DNN-hybrid automatic speech recognition (ASR) systems. Recently, end-to-end (E2E) models have been proposed as an alternative to conventional DNN-hybrid ASR systems. E2E models map a speech signal to a sequence of characters or words using a single neural network, which greatly simplifies the ASR pipeline. However, adaptation of E2E models has received little attention yet. In this paper, we investigate auxiliary feature based adaptation for encoder-decoder E2E models. We employ a recently proposed sequence summary network to compute auxiliary features instead of i-vectors, as it can be easily integrated into E2E models and keep the ASR pipeline simple. Indeed, the sequence summary network allows the auxiliary feature extraction module to be a part of the computational graph of the E2E model. We demonstrate that the proposed adaptation scheme consistently improves recognition performance of three publicly available recognition tasks.

Index Terms: speech recognition, adaptation, end-to-end, auxiliary feature

1. Introduction

Recently end-to-end (E2E) models are becoming a competitive alternative to conventional deep neural network (DNN) hybrid automatic speech recognition (ASR) systems. E2E models replace the acoustic model, lexicon and language model of a conventional ASR system with a single neural network that directly maps an input speech signal to an output sequence of characters or words. E2E systems offer thus the possibility to optimize jointly all components of an ASR system. Moreover, they greatly simplify the ASR training and decoding pipelines.

There are two major approaches for E2E ASR, connectionist temporal classification (CTC) [1, 2] and attention-based encoder-decoder models [3, 4]. A combination of these frameworks has also been proposed recently [5, 6]. These E2E models have recently achieved state-of-the-art performances for languages with a large character set such as Japanese or Chinese [6, 7] or tasks with a large amount of training data [7, 8]. Moreover, these models open the way to new applications such as multi-lingual ASR [9, 10] or direct speech-to-speech translation [11].

E2E models are still relatively novel and many approaches that have been known to be efficient for legacy ASR systems have not been sufficiently investigated yet. For example, acoustic model adaptation is known to improve performance of legacy systems. Indeed, models trained with a large amount of training data are optimized for the average performance over the distribution of the training data, which may not be optimal for a specific condition encountered at test time. Therefore adapting a model to the test conditions is known to improve performance. Conventional adaptation methods include feature transformation [12–14], model retraining [15–17] and auxiliary feature based adaptation [18, 19]. Auxiliary feature based adaptation in particular has recently received a lot of attention as it is an effective way to realize speaker or noise adaptation with a very limited amount of adaptation data.

There have been only a few studies yet on adaptation of E2E systems [7, 20–22]. These works confirm the potential of adaptation for E2E systems. However, they usually make the ASR pipelines more complex as they require either a retraining step [21, 22] or training a separate model to estimate auxiliary features [20] or feature transformations [7]. To the best of our knowledge, auxiliary feature-based speaker adaptation has not been investigated yet for E2E ASR systems.

In this paper we investigate auxiliary feature based speaker adaptation for encoder-decoder models. i-vectors may appear as a natural choice for auxiliary features representing the speakers, given its success with legacy ASR systems [18, 23]. However, it is difficult to integrate the i-vector extraction process within an E2E system as it is hard to express it as a neural network operation that could be integrated into the computational graph of the E2E model. Therefore, we employ instead the recently proposed *sequence summary network* [24] to compute auxiliary features representing speaker characteristics. The sequence summary network consists of a simple feed-forward neural network, whose output is averaged over the duration of the input speech signal so that it is mapped to a single vector representing the speaker. We add the output of the sequence summary network to the input of the encoder to realize speaker adaptation. The sequence summary network can be easily connected to the encoder model and thus trained jointly with the encoder-decoder model. Consequently, the proposed adaptive E2E system keeps the simplicity of a standard E2E ASR pipeline. We call the proposed method *adaptive encoder*. We test our proposed approach on three ASR tasks, i.e. Wall Street Journal (WSJ) [25], TED-LIUM [26] and Corpus of Spontaneous Japanese (CSJ) [27], and demonstrate consistent performance improvements. Moreover, visualization of the auxiliary features confirms that the sequence summary network can effectively capture speaker information.

The remainder of the paper is organized as follows. We review related prior works in Section 2. In Section 3, we briefly describe the baseline E2E system we use in this paper. We then detail the proposed adaptive encoder in Section 4. Finally, we present experimental results in Section 5 and conclude the paper in Section 6.

10.21437/Interspeech.2018-1438

2. Related works

There has been much research on model adaptation of conventional DNN-hybrid acoustic models, including model retraining and its variations [15–17], feature transformation such as feature space maximum likelihood linear regression (fMLLR) or vocal tract length normalization (VTLN) [12–14], and auxiliary feature based adaptation [18, 19].

Model retraining is a very effective approach for acoustic model adaptation when there is enough adaptation data [16, 17, 28, 29]. Retraining-based adaptation has been recently investigated for CTC-based E2E systems to adapt a multilingual ASR system to a target language [21], and for speaker adaptation of multi-microphone encoder-decoder models [22]. Although these works proved that retraining was effective for E2E models as well, the additional retraining step they involve may arguably make the ASR pipeline more complex.

Feature transformations, such as fMLLR [13] or VTLN [12], are also very effective techniques for speaker adaptation when the amount of adaptation data per speaker is relatively large [14]. VTLN has also been shown to be effective for CTC-based E2E systems [7]. However, fMLLR or VTLN exploit a separately trained Gaussian mixture model (GMM) acoustic model to compute the feature transformation. This makes the training and decoding pipelines complex. Moreover, the fMLLR/VTLN computations cannot be easily expressed with neural network components, and therefore their model parameters cannot be learned with error backpropagation, making them difficult to optimize within an E2E framework.

Much research has been performed on auxiliary feature based adaptation, proposing different auxiliary features [18, 19, 30] and model architectures [18, 31, 32]. Auxiliary feature-based adaptation has also been proposed recently for CTC-based multilingual E2E ASR system using an auxiliary feature representing a target language [20]. However, auxiliary feature-based speaker adaptation has not been investigated yet for E2E models. The most widely used approach for speaker adaptation consists of adding i-vectors, which represent the speaker characteristics, to the input of an acoustic model [18]. Such an approach can realize rapid unsupervised adaptation. However, the computation of the i-vector requires a separate GMM model. Therefore, as for fMLLR/VTLN the i-vector extraction process is hard to integrate within an E2E framework. In contrast, sequence summary network has been proposed as an alternative to i-vectors that can be jointly trained with an acoustic model DNN [24]. For hybrid systems, sequence summary network approach achieved comparable performance to i-vectors for speaker adaptation. In addition, the sequence summary network has the advantage that it can be easily integrated into an E2E model, keeping the ASR pipeline simple and allowing the E2E training of the auxiliary feature extraction module. This work is the first study on sequence summary network based adaptation for E2E systems.

3. Attention-based Encoder-decoder ASR

Our baseline E2E model consists of an hybrid CTC/Attention E2E model described in [6] and shown in Fig. 1. In particular, we use the implementation provided in [33]. To simplify the description, we limit the discussion to the encoder-decoder part of the model since it is the only component that we modified. However, note that all experiments were performed using the CTC/Attention hybrid training and decoding scheme.

An attention-based encoder-decoder recognition system receives a T-length input sequence of speech features $X = \{\mathbf{x}_t \in \mathbb{R}^D, t = 1, \ldots, T\}$, and outputs a L-length sequence of characters $W = \{w_l \in \mathbb{U}, l = 1, \ldots, L\}$, where $\mathbb{U}$ is the set of distinct characters. The posterior probability of the output sequence given the observed speech sequence, $p(W|X)$ is obtained as,

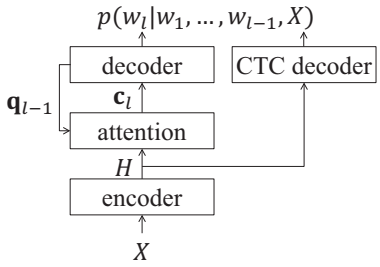

Figure 1: *Schematic diagram of a baseline encoder-decoder model with attention. Refer to [6] for details about the model and the CTC decoding block.*

$$p(W|X) = \prod_{l=1}^{L} p(w_l|w_1, \ldots, w_{l-1}, X), \quad (1)$$

where $p(w_l|w_1, \ldots, w_{l-1}, X)$ is obtained from an encoder-decoder model with attention as described below.

3.1. Encoder

The encoder consists of a neural network that processes all frames of the input sequence and outputs an intermediate representation of the sequence $H = \{\mathbf{h}_t, t = 1, \ldots, T\}$,

$$\mathbf{h}_t = \text{encoder}(\mathbf{x}_t), \quad (2)$$

where here encoder$(\cdot)$ consists of a VGG (very deep convolutional neural network (CNN)) followed by several bidirectional long short-term memory (BLSTM) layers. Note that in practice, the output of the encoder is usually subsampled to reduce the computational complexity [34].

3.2. Attention mechanism

The attention mechanism computes a context vector $\mathbf{c}_l$ to predict the character w_l of the output sequence as,

$$\mathbf{c}_l = \sum_{t=1}^{T} a_{l,t}\mathbf{h}_t \quad (3)$$

$$a_{l,t} = \text{attention}(\mathbf{h}_t, \mathbf{q}_l, \{a_{l-1,t}\}_{t=1}^{T}), \quad (4)$$

where $\{a_{l,t}\}_{t=1}^{T}$ are the attention weights associated with the l^{th} output, and $\mathbf{q}_l$ is an internal state of the decoder recurrent neural network (RNN). We use the location attention mechanism described [34].

3.3. Decoder

Finally, the decoder computes the posterior probability $p(w_l|w_1, \ldots, w_{l-1}, X)$ as,

$$p(w_l|w_1, \ldots, w_{l-1}, X) = \text{decoder}(\mathbf{c}_l, \mathbf{q}_l, w_{l-1}), \quad (5)$$

where decoder$(\cdot)$ consists of an LSTM layer followed by a fully connected layer and a softmax function. The encoder-decoder

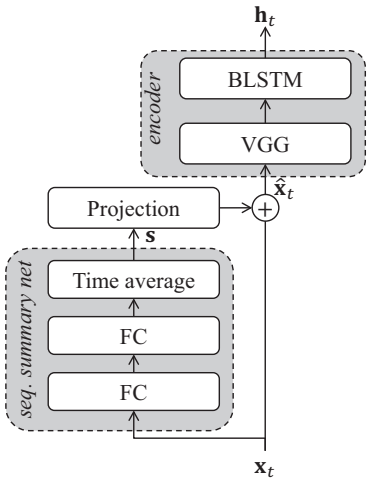

Figure 2: *Schematic diagram of the proposed adaptive encoder with the sequence summary network for the auxiliary feature computation. FC stands for fully connected layer.*

model can model the conditional relation over all input features and past predictions through the state of the decoder RNN $\mathbf{q}_l$. Moreover, all operations can be expressed in as a single neural network, which enables E2E training of all components. However, the current encoder-decoder scheme does not perform speaker adaptation.

4. Proposed adaptive encoder

4.1. Auxiliary input feature based adaptation

Figure 2 is a schematic diagram of the proposed adaptive encoder. Adaptation is realized by adding to the input of the encoder a context dependent bias term derived from the auxiliary feature, $\mathbf{s} \in \mathbb{R}^V$ as,

$$\hat{\mathbf{x}}_t = \mathbf{x}_t + \mathbf{P}\mathbf{s}, \tag{6}$$

where $\hat{\mathbf{x}}_t \in \mathbb{R}^D$ is an adapted input feature and $\mathbf{P} \in \mathbb{R}^{D \times V}$ is a projection matrix used to map the auxiliary feature $\mathbf{s}$ of size V to the dimension, D, of the input feature $\mathbf{x}_t$. Note that adding such a bias is essentially equivalent to concatenating the auxiliary feature to the input of the encoder. Indeed, assuming a fully connected input layer with M hidden units and a transformation matrix, $\mathbf{W} \in \mathbb{R}^{M \times (D+V)}$, we have,

$$\mathbf{W}\begin{bmatrix}\mathbf{x}_t \\ \mathbf{s}\end{bmatrix} = \begin{bmatrix}\mathbf{W}_x & \mathbf{W}_s\end{bmatrix}\begin{bmatrix}\mathbf{x}_t \\ \mathbf{s}\end{bmatrix} = \mathbf{W}_x(\mathbf{x} + \underbrace{\mathbf{W}_x^{-1}\mathbf{W}_s}_{\triangleq \mathbf{P}}\mathbf{s}). \tag{7}$$

Therefore, instead of learning the transformation matrix associated with the auxiliary features $\mathbf{W}_s \in \mathbb{R}^{M \times V}$, we learn $\mathbf{P} \triangleq \mathbf{W}_x^{-1}\mathbf{W}_s \in \mathbb{R}^{D \times V}$ directly. This implementation has the advantage of keeping the configuration of the encoder unchanged, and the implementation for CNN input layers simple.

4.2. Sequence summary network

It is possible to use various types of auxiliary features to represent the context. For example, i-vectors, speaker bottleneck features or noise estimates have been used. Here, to keep the possibility of achieving E2E training, we employ the recently proposed sequence summary network to compute the auxiliary features $\mathbf{s}$.

With the sequence summary network framework, an auxiliary feature $\mathbf{s}$ representing the context of the utterance is computed as,

$$\mathbf{s} = \frac{1}{T}\sum_{t=1}^{T} g(\mathbf{x}_t), \tag{8}$$

where $g(\cdot)$ is a neural network that consists of several fully connected (FC) layers. The time averaging operation reduces the input sequence to a single vector representing its context. In the following experiments, we expect that the context information captured by the auxiliary feature represents speaker information, since we focus on clean speech ASR where the main factor of context variability comes from the speaker characteristics.

Table 1: *Details of the corpora used for the experiments.*

WSJ	
Training set	81 h (283 speakers)
Dev set	1.1 h (5 females, 5 males)
Eval set	0.7 h (3 females, 5 males)
Nb of characters	50
TED-LIUM	
Training set	210 h (5079 talks)
Dev set	1.6 h (1 female, 7 males)
Eval set	2.6 h (2 females, 8 males)
Nb of characters	32
CSJ	
Training set	513 h (3176 lectures)
Test set 1	1.8 h (10 males)
Test set 2	1.9 h (5 females, 5 males)
Test set 3	1.3 h (5 females, 5 males)
Nb of characters	3260

5. Experiments

We performed experiments to confirm the effectiveness of the proposed adaptive encoder on three publicly available corpora, two English language speech corpora WSJ, TED-LIUM and a Japanese corpora, CSJ. The characteristics of the corpora are summarized in Table 1.

5.1. Experimental settings

We used the same model configuration in all three experiments. All parameters were chosen according to the ESPnet recipes, and interested readers should refer to [33, 35] for more details. The baseline E2E system consists of an hybrid CTC/attention system [6]. The input features consists of 80 log mel filterbank coefficients with pitch on each frame. The encoder consists of the two initial blocks of a VGG network followed by six BLSTM layers with 320 units. Down-sampling was performed to reduce the length of the encoded sequence by a factor of 4. We used location based attention mechanism [34]. The decoder consists of a single LSTM layer with 300 units followed by a linear layer with a number of output units corresponding to the number of distinct characters of each task (see Table 1).

The configuration of the sequence summary network is similar to that proposed in [24], i.e. we used three fully connected layers of 512 units with hyperbolic tangent activation functions,

Table 2: *CER [%] for the baseline and proposed adaptive encoder (adapt. enc.), for WSJ, TED-LIUM and CSJ tasks.*

	WSJ		TED-LIUM		CSJ		
	dev	eval	dev	eval	test1	test2	test3
Baseline	7.4	5.5	9.9	10.0	9.8	7.1	7.9
Adapt. enc.	7.1	5.1	9.7	9.7	9.4	6.5	7.2

Table 3: *WER [%] for the baseline and proposed adaptive encoder (adapt. enc.), for WSJ and TED-LIUM tasks with and without RNNLM score combination during decoding.*

	WSJ		TED-LIUM	
	dev	eval	dev	eval
Baseline	21.8	17.3	22.5	22.0
Adapt. enc.	21.3	16.3	21.7	21.1
Baseline +RNNLM	13.2	10.5	-	-
Adapt. enc. +RNNLM	13.2	8.7	-	-

except for the last layer that had linear activation and 100 output units.

All model parameters were randomly initialized. In particular, we did not employ any pre-training technique for the sequence summary network. All models were optimized using the hybrid CTC/attention loss with adadelta for up to 15 epochs.

We used beam search decoding combining the CTC and encoder-decoder outputs, with a beam size of 20. All parameters were tuned on the baseline system and kept unchanged for the proposed adaptive encoder approach. We evaluated all models in terms of character error rate (CER) and word error rate (WER) for the English tasks.

5.2. Results

Table 2 shows the CER for the three tasks using the baseline E2E and the proposed adaptive encoder. The proposed method achieves consistent recognition gains for all test sets. The relative CER improvement ranges from 2 % for the dev set of TED-LIUM to up to more than 8 % for CSJ. These results confirm the potential of the method for speaker adaptation.

English tasks are usually evaluated in terms of WER. Table 3 shows the WER for the WSJ and TED-LIUM tasks. We observed similar relative improvement as in Table 2. We also provide results with RNN language model (LM) score combination during decoding [36] for WSJ (Due to time constraint we could not provide these results for TED-LIUM). The RNNLM consists of a character-based two-layer LSTM LM trained on the WSJ language model training set, which is larger than the speech training set. When using decoding with RNNLM score combination, the proposed method achieved similar performance than the baseline for the dev set, however performance improved significantly for the eval set.

Encoder-decoder models are especially effective for tasks with a large amount of training data or large character set. Performance of encoder-decoder models for English tasks with limited amount of training data such as WSJ and TED-LIUM are still below that of hybrid models. We plan to evaluate our proposed method with larger English data sets in future work. Note that the performance of our baseline on CSJ, which has a large character set, is comparable or slightly superior to conventional hybrid system and representative of the state-of-the art for that task [6]. The relatively large performance improvement observed for CSJ appears thus particularly promising.

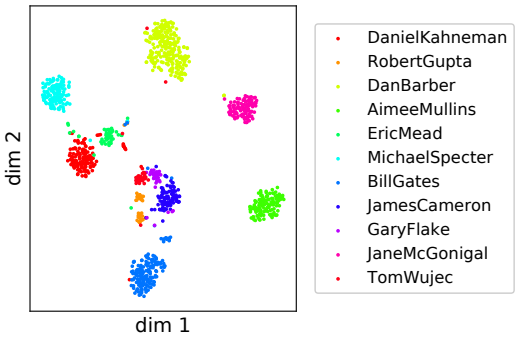

Figure 3: *tSNE plots of the auxiliary features obtained from the sequence summary network for the utterances of the eval set of the TED-LIUM corpus. The colors represent the different speakers. (Best in colors)*

5.3. Discussion

We confirmed that the sequence summary network extracts speaker information by looking at the auxiliary features ($\mathbf{s}$ in Eq. (8)) for the different utterances of the eval set of the TED-LIUM corpus. Figure 3 plots a 2D representation of the auxiliary features obtained by reducing the dimensionality using tSNE [37]. Note that similar plots were observed for the other tasks but were omitted for space consideration. We observe clearly that auxiliary features corresponding to the same speaker are clustered together, which confirms that the sequence summary network learns to extract speaker information.

In this paper, we employed an encoder that includes a BLSTM layer, which can potentially capture long context such as speaker information. One could argue that since both the sequence summary network and the encoder see the same information, there is no need for the adaptation module. However, the sequence summary network performs an averaging over all input features and is thus specially designed to capture the overall context of the input signal, in this case the speaker information. The performance improvement observed in the experiments confirms the benefit of using such a dedicated module to capture the overall context information and perform adaptation.

6. Conclusions

This paper demonstrates the effectiveness of auxiliary feature-based adaptation for end-to-end ASR systems. We proposed using a sequence summary network to learn speaker representations within an end-to-end ASR framework. We confirmed the effectiveness of the proposed method with three ASR corpora showing consistent gains. Especially, we could achieve large performance improvement over a strong baseline recognizer for Japanese.

Future work will include investigations with larger English corpora, other approaches for exploiting the auxiliary features [31, 32] as well as integration into an online decoding scheme [38].

7. References

[1] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proc. of ICML'06*, 2006, pp. 369–376.

[2] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proc. of ICML'14*, 2014, pp. 1764–1772.

[3] J. Chorowski, D. Bahdanau, K. Cho, and Y. Bengio, "End-to-end continuous speech recognition using attention-based recurrent NN: First results," *arXiv preprint arXiv:1412.1602*, 2014.

[4] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Proc. of ICASSP'15*, 2015, pp. 4960–4964.

[5] S. Kim, T. Hori, and S. Watanabe, "Joint CTC-attention based end-to-end speech recognition using multi-task learning," in *Proc. of ICASSP'17*, 2017, pp. 4835–4839.

[6] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid CTC/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.

[7] Y. Miao, M. Gowayyed, X. Na, T. Ko, F. Metze, and A. H. Waibel, "An empirical exploration of CTC acoustic models," in *Proc. of ICASSP'16*, 2016, pp. 2623–2627.

[8] C.-C. Chiu, T. N. Sainath, Y. Wu, R. Prabhavalkar, P. Nguyen, Z. Chen, A. Kannan, R. J. Weiss, K. Rao, E. Gonina, N. Jaitly, B. Li, J. Chorowski, and M. Bacchiani, "State-of-the-art speech recognition with sequence-to-sequence models," in *Proc. of ICASSP'18*, 2018.

[9] S. Watanabe, T. Hori, and J. R. Hershey, "Language independent end-to-end architecture for joint language identification and speech recognition," in *Proc. of ASRU'17*, 2017, pp. 265–271.

[10] B. Li, T. N. Sainath, K. C. Sim, M. Bacchiani, E. Weinstein, P. Nguyen, Z. Chen, Y. Wu, and K. Rao, "Multi-dialect speech recognition with a single sequence-to-sequence model," in *Proc. of ICASSP'18*, 2018.

[11] A. Berard, O. Pietquin, C. Servan, and L. Besacier, "Listen and translate: A proof of concept for end-to-end speech-to-text translation," in *Proc. of NIPS workshop on End-to-end Learning for Speech and Audio Processing*, 2016.

[12] J. Cohen, T. Kamm, and A. G. Andreou, "Vocal tract normalization in speech recognition: Compensating for systematic speaker variability," *The Journal of the Acoustical Society of America*, vol. 97, pp. 3246–3247, 1995.

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

[14] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Proc. of ASRU'11*, 2011, pp. 24–29.

[15] J. Neto, L. Almeida, M. Hochberg, C. Martins, L. Nunes, S. Renals, and T. Robinson, "Speaker-adaptation for hybrid HMM-ANN continuous speech recognition system," in *Proc. of EUROSPEECH'95*, 1995, pp. 2171–2174.

[16] H. Liao, "Speaker adaptation of context dependent deep neural networks," in *Proc. of ICASSP'13*, 2013, pp. 7947–7951.

[17] P. Swietojanski and S. Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," in *Proc of SLT'14*, 2014, pp. 171–176.

[18] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *Proc. of ASRU'13*, 2013, pp. 55–59.

[19] M. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. of ICASSP'13*, 2013, pp. 7398–7402.

[20] M. Müller, S. Stüker, and A. Waibel, "Language adaptive multilingual CTC speech recognition," in *Speech and Computer*, A. Karpov, R. Potapova, and I. Mporas, Eds. Springer International Publishing, 2017, pp. 473–482.

[21] S. Tong, P. N. Garner, and H. Bourlard, "Multilingual training and cross-lingual adaptation on CTC-based acoustic model," *arxiv*, vol. abs/1711.10025, 2017.

[22] T. Ochiai, S. Watanabe, S. Katagiri, T. Hori, and J. Hershey, "Speaker adaptation for multichannel end-to-end speech recognition," in *Proc. of ICASSP'18*, 2018.

[23] M. Karafiat, L. Burget, P. Matejka, O. Glembek, and J. Cernocky, "ivector-based discriminative adaptation for automatic speech recognition," in *Proc. of ASRU'11*, 2011, pp. 152–157.

[24] K. Vesely, S. Watanabe, K. Zmolikova, M. Karafiat, L. Burget, and J. H. Cernocky, "Sequence summarizing neural network for speaker adaptation," in *Proc. of ICASSP'16*, 2016, pp. 5315–5319.

[25] D. B. Paul and J. M. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Proc. SNL'92*. Morristown, NJ, USA: Association for Computational Linguistics, 1992, pp. 357–362.

[26] A. Rousseau, P. Delglise, and Y. Estve, "TED-LIUM: an automatic speech recognition dedicated corpus," in *Proc. of LREC12*, 2012, pp. 125–129.

[27] K. Maekawa, H. Koiso, S. Furui, and I. H., "Spontaneous speech corpus of Japanese," in *Proc. of LREC'00*, 2000, pp. 947–952.

[28] R. Gemello, F. Mana, S. Scanzio, P. Laface, and R. de Mori, "Adaptation of hybrid ANN/HMM models using linear hidden transformations and conservative training," in *Proc. of ICASSP'06*, vol. 1, 2006, pp. 1189–1192.

[29] D. Yu, K. Yao, H. Su, G. Li, and F. Seide, "KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition," in *Proc. of ICASSP'13*, 2013, pp. 7893–7897.

[30] H. Huang and K. C. Sim, "An investigation of augmenting speaker representations to improve speaker normalisation for dnn-based speech recognition," in *Proc. of ICASSP'15*, 2015, pp. 4610–4613.

[31] M. Delcroix, K. Kinoshita, A. Ogawa, C. Huemmer, and T. Nakatani, "Context adaptive neural network based acoustic models for rapid adaptation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 5, pp. 895–908, May 2018.

[32] L. Samarakoon and K. C. Sim, "Subspace LHUC for fast adaptation of deep neural network acoustic models," in *Proc. of Interspeech'16*, 2016, pp. 1593–1597.

[33] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, J. Heymann, M. Wiesner, N. Chen, and N. E. Y. Soplin, "ESPnet: End-to-end speech processing toolkit," in *Proc. of Interspeech'18 (Submitted)*, 2018.

[34] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. of NIPS'15*, 2015, pp. 577–585.

[35] "End-to-End Speech Processing Toolkit," *https://github.com/espnet/espnet*, cited March 21 2018.

[36] T. Hori, S. Watanabe, Y. Zhang, and W. Chan, "Advances in joint CTC-attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM," in *Interspeech*, 2017, pp. 949–953.

[37] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.

[38] T. Ochiai, M. Delcroix, K. Kinoshita, A. Ogawa, T. Asami, S. Katagiri, and T. Nakatani, "Cumulative moving averaged bottleneck speaker vectors for online speaker adaptation of CNN-based acoustic models," in *Proc of ICASSP'17*, 2017, pp. 5175–5179.