Selective Adaptation of End-to-End Speech Recognition using Hybrid CTC/Attention Architecture for Noise Robustness

Cong-Thanh Do
Toshiba Cambridge Research Laboratory
Cambridge, UK
cong-thanh.do@crl.toshiba.co.uk

Shucong Zhang
The University of Edinburgh
Edinburgh, UK
s1603602@sms.ed.ac.uk

Thomas Hain
The University of Sheffield
Sheffield, UK
t.hain@sheffield.ac.uk

***Abstract*—This paper investigates supervised adaptation of end-to-end speech recognition, which uses hybrid connectionist temporal classification (CTC)/Attention architecture, for noise robustness. The components of the architecture, namely the shared encoder, the attention decoder's long short-term memory (LSTM) layers, and the soft-max layers of the CTC part and attention part, are adapted separately or together using limited amount of adaptation data. When adapting the shared encoder, we propose to adapt only the connections of the memory cells in the memory blocks of bidirectional LSTM (BLSTM) layers to improve performance and reduce the time for adapting the models. In within-domain and cross-domain adaptation scenarios, experimental results show that adaptation of end-to-end speech recognition using the hybrid CTC/Attention architecture is effective even when the amount of adaptation data is limited. In cross-domain adaptation, substantial performance improvement can be achieved with only 2.4 minutes of adaptation data. In both adaptation scenarios, adapting only the memory cells of the BLSTM layers in the shared encoder yields comparable or slightly better performance while yielding smaller adaptation time than the adaptation of other components or the whole architecture, especially when the amount of adaptation data is less than or equal to 10 minutes.**

***Index Terms*—End-to-end speech recognition, noise robustness, adaptation, connectionist temporal classification, attention**

I. Introduction

Noise robustness is a necessary quality of end-to-end automatic speech recognition (ASR) systems working in realistic environments. End-to-end ASR uses a single neural network architecture within a deep learning framework to perform speech-to-text task. Major approaches for end-to-end ASR includes: attention-based approach [1] uses an attention mechanism to create required alignments between acoustic frames and output symbols which have different lengths, connectionist temporal classification (CTC) approach [2] uses Markov assumptions to address sequential problem by dynamic programming, and the transformer-based approach [3] learns sequential information via a self-attention mechanism instead of recurrent connections.

For noise robustness, multi-style training is often used to train speech models with a multi-condition training set to cover a wide range of application environments. However, it is hard to cover all of the possible noise types and signal-to-noise ratios (SNRs) that may be present in future test environments. Adaptation, among others approaches [4], can be used to adapt the end-to-end models to a specific environment to close the gap between training and test.

The hybrid CTC/Attention architecture is one of the architectures for end-to-end ASR which could provide state-of-the-art speech recognition performance [5], [6]. This architecture combines the advantages of CTC-based and attention-based encoder-decoder architectures in training and decoding. As a result of this combination, the hybrid CTC/Attention architecture consists of different components, and each component has several layers. Therefore, adapting this architecture could be costly as sufficient time and adaptation data would be required for effective adaptation. When adaptation data is limited, adapting all the components of the hybrid CTC/Attention architecture may not be the most effective.

In this work, we investigate the supervised adaptation of the hybrid CTC/Attention end-to-end ASR for noise robustness. We selectively adapt each component of the architecture to investigate which component is the best to adapt when the amount of adaptation data is limited. Two evaluation scenarios, namely within-domain and cross-domain adaptations, are established. These two evaluation scenarios use training, adaptation, and test data from the CHiME-4 and Aurora-4 corpora which were designed for noise robust ASR tasks. We show that adapting only the matrices and vectors that connect the memory cells in the memory blocks of the BLSTM layers [7], [8] with the network is effective both in terms of improving performance and reducing the time for adapting the models, compared to when adapting the whole hybrid CTC/Attention architecture.

The paper is organized as follows. Section II presents related works. The adaptation of the hybrid CTC/Attention architecture are presented in section III. ASR experiments are presented in section IV which includes the experimental setup and results. Finally, section V concludes the paper.

II. Related works

A number of studies have been published in the literature on adaptation in end-to-end speech recognition [9]–[13]. In [9], domain adaptation techniques were applied to adapt hybrid

CTC/Attention end-to-end ASR systems. Adaptation techniques, for instance cluster adaptive training (CAT), factorized hidden layer (FHL) adaptation, or domain specific gating were applied when three hours of unlabeled adaptation data is available [9]. In [9], the CAT and FHL adaptations were applied to the projections of the BLSTM layers. There was however no specific investigation on which component of the hybrid CTC/Attention architecture is the most effective for adaptation to gain noise robustness when a limited amount of labeled adaptation data is available. This investigation will be carried out in the present work.

III. Adaptation of hybrid CTC/Attention end-to-end speech recognition

A. Hybrid CTC/Attention architecture

The hybrid CTC/Attention architecture for end-to-end ASR is depicted in Fig. 1. The T-length acoustic feature sequence $X = \{\mathbf{x}_t \in \mathbb{R}^d | t = 1, ..., T\}$ is taken as input and the L-length character sequence $C = \{c_l \in \mathcal{U} | l = 1, ..., L\}$ is produced at the output. Here, $\mathbf{x}_t$ is a d-dimensional feature vector at frame t and $\mathcal{U}$ is a set of distinct characters. During training, a CTC objective function $p_{ctc}(C|X)$ is used as an auxiliary task to train the attention model encoder within the multiobjective learning (MOL) framework. The objective to be maximized, $\mathcal{O}_{MOL}(\theta)$, is a logarithmic linear combination of the CTC objective, $p_{ctc}(C|X)$, and the attention objective, $p_{att}(C|X)$:

$$\mathcal{O}_{MOL}(\theta) = \lambda \log p_{ctc}(C|X) + (1-\lambda) \log p_{att}(C|X) \quad (1)$$

where λ is a tunable parameter which satisfies $0 \leq \lambda \leq 1$, and θ represents all the parameters of the objective function $\mathcal{O}_{MOL}$. The CTC objective $p_{ctc}(C|X)$ is defined as [5]:

$$p_{ctc}(C|X) = \sum_{Z} \prod_{t=1}^{T} p(z_t|z_{t-1}, C) p(z_t|X), \quad (2)$$

where $Z = \{z_t \in \mathcal{U} \cup \{< b >\} | t = 1, ..., T\}$ is a framewise character sequence with an additional blank symbol $< b >$ [2]. The attention objective $p_{att}(C|X)$ is defined as:

$$p_{att}(C|X) = \prod_{l=1}^{L} p(c_l|c_1^*, ..., c_{l-1}^*, X), \quad (3)$$

where $c_1^*, ..., c_{l-1}^*$ are the ground truths of the characters $c_1, ..., c_{l-1}$ which are output prior to the character c_l [5].

The objective function $\mathcal{O}_{MOL}(\theta)$ is used to train the hybrid CTC/Attention architecture using back-propagation algorithm. The architecture consists of a shared encoder, a joint decoder, and soft-max layers. The shared encoder consists of deep convolutional neural network (CNN) layers [14] followed by BLSTM layers. The attention decoder is usually a LSTM neural network. There are two soft-max layers corresponding to the CTC part and attention part which transform the hidden activations from the shared encoder and the attention decoder into posterior probabilities for computing the CTC and attention-based scores, respectively [5], [6].

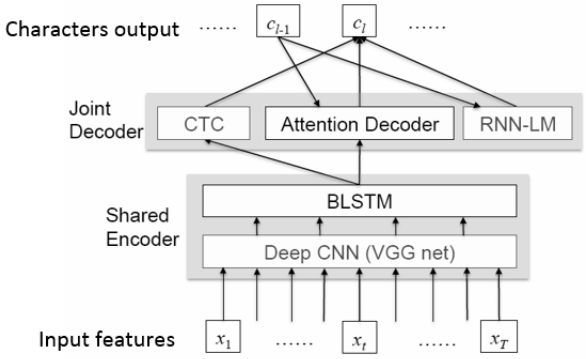

Fig. 1: Hybrid CTC/attention architecture for end-to-end speech recognition [5].

B. Gradient-based adaptation

Given a hybrid CTC/Attention end-to-end speech recognition system which is trained on the training data, it can be adapted to better fit the test environments with available adaptation data which characterize the new environments. In this paper, supervised adaptation is studied, i.e. the transcription of the adaptation data is available. During adaptation, back-propagation algorithm is used to fine-tune the whole architecture, or a specific component of the architecture, with respect to the objective function $\mathcal{O}_{MOL}(\theta)$, using the adaptation data. During the minimization of the objective function using stochastic gradient descent [15], the parameters θ of $\mathcal{O}_{MOL}(\theta)$ are updated as:

$$\theta = \theta - \eta \nabla_\theta [\mathcal{O}_{MOL}(\theta)], \quad (4)$$

where η is the learning rate and $\nabla_\theta [\mathcal{O}_{MOL}(\theta)]$ is the gradient of $\mathcal{O}_{MOL}(\theta)$ with respect to the parameters θ. If only a specific component of the architecture is adapted, the gradients of the other components' parameters are set to zero. Therefore, only the gradient of the parameters of the specific component contributes to the overall gradient $\nabla_\theta [\mathcal{O}_{MOL}(\theta)]$ for updating θ as in equation (4), and hence, only the parameters of this component are updated.

C. Adaptation of BLSTM memory cells

In BLSTM network, each BLSTM layer consists of a forward uni-directional LSTM layer and a backward uni-directional LSTM layer [8]. A LSTM layer consists of a set of memory blocks which are recurrently connected. Each memory block contains one or more recurrently connected memory cell and three multiplicative unites - the input, output, and forget gates - that provide continuous analogue of write, read, and reset operations for the memory cells [7]. The network can only interact with the memory cell via the gates. In the present work, each memory block has one memory cell and peephole connections are not used within memory blocks. The connections within the LSTM memory blocks can be described as follows:

$$\mathbf{f}_t = \sigma_g(\mathbf{W}_f \mathbf{y}_t + \mathbf{U}_f \mathbf{h}_{t-1} + \mathbf{b}_f) \quad (5)$$

$$\mathbf{i}_t = \sigma_g(\mathbf{W}_i \mathbf{y}_t + \mathbf{U}_i \mathbf{h}_{t-1} + \mathbf{b}_i) \quad (6)$$

$$\mathbf{o}_t = \sigma_g(\mathbf{W}_o\mathbf{y}_t + \mathbf{U}_o\mathbf{h}_{t-1} + \mathbf{b}_o) \quad (7)$$

$$\mathbf{c}_t = \mathbf{f}_t \circ \mathbf{c}_{t-1} + \mathbf{i}_t \circ \rho_c(\mathbf{W}_c\mathbf{x}_t + \mathbf{U}_c\mathbf{h}_{t-1} + \mathbf{b}_c) \quad (8)$$

$$\mathbf{h}_t = \mathbf{o}_t \circ \rho_h(\mathbf{c}_t), \quad (9)$$

where $\mathbf{f}_t$, $\mathbf{i}_t$, and $\mathbf{o}_t$ are the activation vectors of the forget, input, and output gates, respectively. The vectors $\mathbf{y}_t$ and $\mathbf{h}_t$ are the static input and recurrent input (or output) vectors of the memory blocks, respectively, and $\mathbf{c}_t$ is the cell state vector. The matrices $\mathbf{W}$ and $\mathbf{U}$ are the weight matrices which connect the static and recurrent inputs, respectively, to gates and memory cells. The operator $\circ$ denotes the element-wise product and the vectors $\mathbf{b}$ are bias vectors. σ_g is a sigmoid function whereas ρ_c and ρ_h are hyperbolic tangent functions.

In [16], the matrices $\mathbf{W}$ and $\mathbf{U}$ of the gates and memory cells were used for speaker adaptation in hybrid hidden Markov model (HMM) - deep neural network (DNN) ASR. Adapting the matrices related to the LSTM memory cells was the most effective compared to adapting those related to the gates. Indeed, the memory cell is the central component of a LSTM memory block. In this work on the adaptation of the hybrid CTC/Attention architecture for end-to-end ASR, we thus propose to adapt the matrices $\mathbf{W}_c$, $\mathbf{U}_c$, and vector $\mathbf{b}_c$ (see equation (8)) which model the connections of the memory cells with the BLSTM network. Adapting only the memory cells instead of the entire BLSTM layers could make adaptation more efficient when the adaptation data is limited because adapting all the connections would need sufficient amount of adaptation data. In addition, adapting only the memory cells could reduce the time for doing adaptation because less connections are adapted. For the experiments in this paper, we adapt the memory cells in both the forward and backward LSTM layers, for all the BLSTM layers.

IV. Experiments

A. Experimental setup

Two adaptation scenarios are set up: within-domain and cross-domain adaptations. A hybrid CTC/Attention end-to-end ASR system is trained on the multi-condition training data of CHiME-4 corpus [17] which consists of around 189 hours of speech. The training is done using the ESPnet toolkit [6] and PyTorch [18]. This system is used in both adaptation scenarios. The CHiME-4 multi-condition training data consists of the clean speech utterances from WSJ corpus and simulated and real noisy data. The real data consists of 6-channel recordings of sentences from WSJ corpus spoken live in four environments: café, street junction, public transport (bus), and pedestrian area. The simulated data was constructed by mixing WSJ clean utterances into environment background recordings from the four mentioned environments. All the data were sampled at 16 kHz. Audio recorded from all the microphone channels are included in the CHiME-4 multi-condition training data, named `tr05_multi_noisy_si284` in the ESPnet CHiME-4 recipe. The `dt05_multi_isolated_1ch_track` set was used as the validation set during the training.

For the within-domain adaptation experiment, data from the CHiME-4 evaluation set `et05_real_isolated_1ch_track` is used. This set consists of 1320 utterances recorded in the four mentioned real noisy environments using a single microphone. The environments in which these utterances were recorded are similar but not identical to those in which the simulated and real noisy data of the multi-condition training set were produced. We randomly separate 300 utterances from this set for adaptation and create a new evaluation set with the 1020 remaining utterances. The 300 utterances are step-by-step reduced to create smaller adaptation sets of 200, 100, 80, 60, 40, 20 utterances (see Tab. I). These sets are used as data to adapt the system trained on the CHiME-4 multi-condition training data set. The evaluation is done on the new evaluation set of 1020 utterances. There is no significant difference in CER and WER while using the new evaluation set of 1020 utterances and the original set of 1320 utterances.

TABLE I: Equivalence between the number of utterances and the total length of speech for two adaptation scenarios.

Number of utterances	Total length (in minutes)	
	Within-domain	Cross-domain
20	1.9	2.4
40	4.3	5.1
60	6.5	8.0
80	8.5	10.3
100	10.1	12.8
200	19.8	25.2
300	29.8	37.7

For the cross-domain adaptation experiment, data from 14 test sets of Aurora-4 are used. The 14 test sets of Aurora-4 were created by corrupting two clean test sets, recorded by a primary Sennheiser microphone and a secondary microphone, with six types of noises: airport, babble, car, restaurant, street, and train, at 5-15 dB SNRs. The two clean test sets were also included in the 14 test sets. The noises in Aurora-4 are different from those in the CHiME-4 multi-condition training data. From the 14 test sets of Aurora-4, 1400 utterances are randomly selected, 100 from each test set. Similar to the within-domain adaptation, 300 utterances are randomly separated from the set of 1400 utterances. From these 300 utterances, similar protocol is applied to create smaller adaptation sets (see Tab. I). The evaluation is done on the new evaluation set of 1100 remaining utterances.

B. Implementation details

The shared encoder of the hybrid CTC/Attention architecture is made up of initial layers of the VGG net architecture (deep CNN) [19] followed by a 4-layer pyramid BLSTM (BLSTM with subsampling [6]). We use a 6-layer CNN architecture which consists of two consecutive 2D convolutional layers followed by one 2D Max-pooling layer, then another two 2D convolutional layers followed by one 2D max-pooling layer. The 2D filters used in the convolutional layers have the same size of 3×3. The max-pooling layers have patch of 3×3 and stride of 2×2. The 4-layer BLSTM has 1024 memory blocks in each layer and direction, and linear projection is followed by each BLSTM layer. The subsampling factor performed by the BLSTM is 4 [6].

Location-based attention mechanism [1] is used. This mechanism uses 10 centered convolution filters of width 100 to extract the convolutional features. The attention decoder network is a 1-layer LSTM with 1024 memory blocks. The training is performed with 20 epochs using PyTorch. The acoustic features are 40-dimensional Mel filter-bank features which are augmented with 3-dimensional pitch features [6]. The AdaDelta algorithm [20] is used for the optimization. During training, λ is set to 0.5 to be consistent with the ESPnet training recipe for CHiME-4 [6].

During joint decoding, CTC and attention-based scores are combined in a one-pass beam search algorithm [6]. A recurrent neural network language model (RNN-LM), which is a 1-layer LSTM, is trained on the transcriptions of the training data. This RNN-LM is used in the joint decoding where its log probability is combined with the CTC and attention scores [6]. The weight of the RNN-LM's log probability is set to 0.1 and the beam width is set to 20 during decoding.

C. Results

1) Within-domain adaptation: Tabs. II and III show the CERs and WERs, respectively, when all the components or a specific component of the hybrid CTC/Attention architecture are adapted with various amounts of adaptation data. In these Tables, A denotes the adaptation of all the components of the architecture. B, C, and D denote the adaptation of the shared encoder, the attention decoder's LSTM layers, and the soft-max layers, respectively. Besides, E denotes the adaptation of the CNN and the BLSTM memory cells (see section III-C).

In this scenario, adapting all the components or a specific component of the architecture helps reducing the CERs and WERs. Among the adaptations of specific components, adapting CNN + BLSTM memory cells yields on average the best CER reduction. In general, the more adaptation data is used, the larger the CER reduction is obtained. Compared to the baseline without adaptation, adapting all the components and adapting the CNN + BLSTM memory cells yield 11% and 11.5% relative CER reductions, respectively, with 10 minutes of adaptation data, and yield 23.9% and 17.3% relative CER reductions, respectively, with about 30 minutes of adaptation data (300 utterances).

Fig. 2 shows the relative reduction of the time used for adapting specific components compared to that used for adapting all the components. The time for adapting the model, or adaptation time, is calculated as the total time used by an adaptation process running on the same machine. The data reading/writing times are excluded. It can be observed that adapting CNN + BLSTM memory cells yields on average 22.7% relative reduction of adaptation time, compared to adapting all the components, while yielding similar CER reduction when the amount of adaptation data is less than or equal to 10 minutes.

2) Cross-domain adaptation: Tabs. IV and V show the CERs and WERs for the cross-domain adaptation experiments. Adapting the shared encoder (B) and adapting the CNN + BLSTM memory cells (E) yield, on average, slightly better

TABLE II: CERs with within-domain adaptation. The CER of the baseline system without adaptation equals 19.1%.

No. of utts. \ Adaptation	A	B	C	D	E
20	19.1	18.7	18.4	18.9	19.0
40	18.1	18.7	18.5	18.7	18.5
60	17.5	18.1	18.2	18.8	17.7
80	17.9	17.3	18.0	19.2	17.4
100 ($\approx$ 10 minutes)	17.0	17.8	17.8	19.1	16.9
200	16.1	15.9	17.0	19.1	16.4
300	14.7	15.2	16.5	19.0	15.8
Average (All)	17.2	17.4	17.8	19.0	17.4
Average ($\leq$ 10 minutes)	17.9	18.1	18.2	18.9	17.9

TABLE III: WERs with within-domain adaptation. The WER of the baseline system without adaptation is 32.4%.

No. of utts. \ Adaptation	A	B	C	D	E
20	32.4	32.3	31.8	32.0	32.5
40	30.9	32.1	31.8	31.6	31.9
60	30.1	31.6	31.3	31.4	30.3
80	30.8	30.3	30.9	31.8	30.1
100 ($\approx$ 10 minutes)	29.3	30.4	30.6	31.7	29.4
200	27.6	27.7	29.1	31.1	28.3
300	25.2	26.8	28.4	30.9	27.6
Average (All)	29.5	30.2	30.6	31.5	30.0
Average ($\leq$ 10 minutes)	30.7	31.3	31.3	31.7	30.8

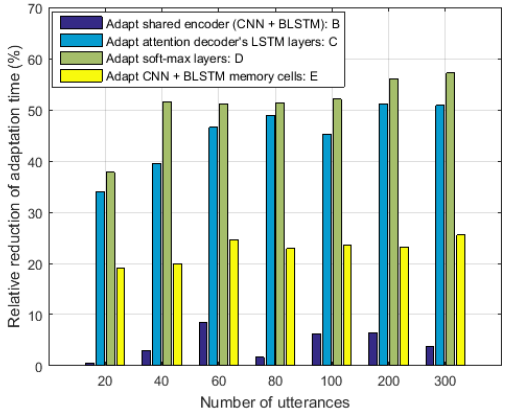

Fig. 2: Adaptation time reductions obtained when adapting specific components compared to when adapting all the components of the architecture (A), in within-domain adaptation.

CER and WER reductions compared to adapting all the components, especially when the amount of adaptation data is less than or equal to 10 minutes. With only 20 utterances (2.4 minutes of speech) for adaptation, up to 12.8% relative CER reductions are obtained with these adaptations. Compared to the baseline without adaptation, adapting all the components and adapting the CNN + BLSTM memory cells yield 27.9% and 31.4% relative CER reductions, respectively, with 10 minutes of adaptation data, and yield 61.6% and 57.0% relative CER reductions, respectively, with about 38 minutes of adaptation data (300 utterances). Similar patterns on the reduction of adaptation time can be observed in Fig. 3 where adapting only the CNN + BLSTM memory cells yields on average 24.1% relative adaptation time reduction.

TABLE IV: CERs with cross-domain adaptation. The CER of the baseline system without adaptation is 8.6%.

No. of utts. \ Adaptation	A	B	C	D	E
20	7.5	7.7	8.2	8.1	7.5
40	7.5	6.9	8.1	8.2	6.9
60	6.6	6.3	7.9	8.1	6.3
80 ($\approx$ 10 minutes)	6.2	6.3	7.8	8.0	5.9
100	5.8	5.4	7.7	7.8	5.9
200	4.2	4.2	7.5	7.8	4.5
300	3.3	3.1	6.9	7.5	3.7
Average (All)	5.9	5.7	7.7	7.9	5.8
Average ($\leq$ 10 minutes)	7.0	6.8	8.0	8.1	6.7

TABLE V: WERs with cross-domain adaptation. The WER of the baseline system without adaptation is 15.5%.

No. of utts. \ Adaptation	A	B	C	D	E
20	13.9	14.4	15.3	14.9	14.3
40	13.8	12.6	15.0	14.9	12.9
60	12.2	11.6	14.6	14.5	11.8
80 ($\approx$ 10 minutes)	11.5	11.7	14.4	14.4	11.1
100	10.6	10.3	14.1	14.1	11.0
200	8.1	8.2	13.4	13.5	8.8
300	6.5	6.1	12.3	13.2	7.2
Average (All)	10.9	10.7	14.2	14.2	11.0
Average ($\leq$ 10 minutes)	12.9	12.6	14.8	14.7	12.5

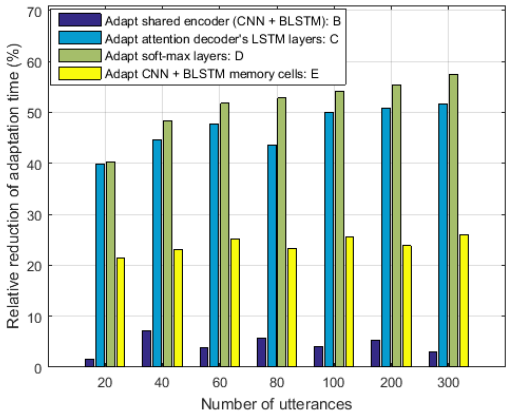

Fig. 3: Adaptation time reductions obtained when adapting specific components compared to when adapting all the components of the architecture (A), in cross-domain adaptation.

V. Conclusion

Supervised adaptation was investigated in end-to-end speech recognition, which uses the hybrid CTC/Attention architecture, for noise robustness. The adaptations of a specific component or all the components of the architecture were examined. Experimental results have shown that adaptation of end-to-end speech recognition is effective even when the amount of adaptation data is limited. In cross-domain adaptation, substantial performance improvement could be achieved with only 2.4 minutes of adaptation data. In both within-domain and cross-domain adaptations, adapting the CNN and the memory cells in the memory blocks of the BLSTM layers yielded comparable or slightly better performance while yielding smaller adaptation time than the adaptation of other components or the whole architecture, especially when the amount of adaptation data is less than or equal to 10 minutes.

Acknowledgment

The authors would like to thank Dr. Rama Doddipatla (Toshiba) for valuable discussions and comments on the study.

References

[1] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. Advances in Neural Information Processing Systems (NIPS)*, 2015, pp. 577–585.

[2] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proc. of the 31st International Conference on Machine Learning*, Beijing, China, June 2014, pp. 1764–1772.

[3] L. Dong, S. Xu, and B. Xu, "Speech-transformer: A no-recurrence sequence-to-sequence model for speech recognition," in *Proc. IEEE ICASSP*, Calgary, Canada, April 2018, pp. 5884–5888.

[4] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An overview of noise-robust automatic speech recognition," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 22, pp. 745–777, April 2014.

[5] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid CTC/Attention architecture for end-to-end speech recognition," *IEEE J. Sel. Topics Signal Process.*, vol. 11, pp. 1240–1253, December 2017.

[6] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: end-to-end speech processing toolkit," in *Proc. INTERSPEECH*, Hyderabad, India, September 2018, pp. 2207–2211.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735–1780, November 1997.

[8] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM networks," in *Proc. of Int. Joint Conf. on Neural Networks*, Montreal, Canada, July-August 2005, pp. 2047–2052.

[9] L. Samarakoon, B. Mak, and A. Y. S. Lam, "Domain adaptation of end-to-end speech recognition in low-resource settings," in *Proc. IEEE Spoken Language Technology Workshop*, Athens, Greece, December 2018, pp. 382–388.

[10] T. Ochiai, S. Watanabe, S. Katagiri, T. Hori, and J. Hershey, "Speaker adaptation for multichannel end-to-end speech recognition," in *Proc. IEEE ICASSP*, Calgary, Canada, December 2018, pp. 6707–6711.

[11] M. Delcroix, S. Watanabe, A. Ogawa, S. Karita, and T. Nakatani, "Auxilary feature based adaptation of end-to-end ASR systems," in *Proc. INTERSPEECH*, Hyderabad, India, September 2018, pp. 2444–2448.

[12] Z. Meng, Y. Gaur, J. Li, and Y. Gong, "Speaker adaptation for attention-based end-to-end speech recognition," in *Proc. INTERSPEECH*, Graz, Austria, September 2019, pp. 241–245.

[13] E. Tsunoo, Y. Kashiwagi, S. Asakawa, and T. Kumakura, "End-to-end adaptation with backpropagation through WFST for on-device speech recognition system," in *Proc. INTERSPEECH*, Graz, Austria, September 2019, pp. 764–768.

[14] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," in *The Handbook of Brain Theory and Neural Networks*. MIT Press, 1995.

[15] S. Ruder, "An overview of gradient descent optimisation algorithms," in *arXiv preprint arXiv: 1609.04747*, 2016.

[16] C. Liu, Y. Wang, K. Kumar, and Y. Gong, "Investigations on speaker adaptation of LSTM RNN models for speech recognition," in *Proc. IEEE ICASSP*, Shanghai, China, May 2016, pp. 5020–5024.

[17] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third CHIME speech separation and recognition challenge: dataset, tasks and baselines," in *Proc. IEEE ASRU*, AZ, USA, Dec. 2015, pp. 504–511.

[18] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *NeurIPS*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., pp. 8024–8035. Curran Associates, Inc., 2019.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015.

[20] M. D. Zeiler, "Adadelta: an adaptive learning rate method," in *arXiv preprint arXiv: 1212.5701*, 2012.