INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Systems for Low-Resource Speech Recognition Tasks in Open Automatic Speech Recognition and Formosa Speech Recognition Challenges

Hung-Pang Lin, Yu-Jia Zhang, Chia-Ping Chen

National Sun Yat-Sen University, Taiwan

m083040013@g-mail.nsysu.edu.tw, m083040025@nsysu.edu.tw, cpchen@cse.nsysu.edu.tw

Abstract

We, in the team name of NSYSU-MITLab, have participated in low-resource speech recognition of the Open Automatic Speech Recognition Challenge 2020 (OpenASR20) and Formosa Speech Recognition Challenge 2020 (FSR-2020). For the tasks in the challenges, we build and compare end-to-end (E2E) systems and Deep Neural Network Hidden Markov Model (DNN-HMM) systems. In E2E systems, we implement an encoder with Conformer architecture and a decoder with Transformer architecture. In addition, a speaker classifier with a gradient reversal layer is included in the training phase to improve the robustness to speaker variation. In DNN-HMM systems, we implement the Time-Restricted Self-Attention and Factorized Time Delay Neural Networks for the DNN front-end acoustic representation learning. In OpenASR20, the best word error rates we achieved are 61.45% for Cantonese and 74.61% for Vietnamese. In FSR-2020, the best character error rate we achieved is 43.4% for Taiwanese Southern Min Recommended Characters and the best syllable error rate is 25.4% for Taiwan Minnanyu Luomazi Pinyin.

Index Terms: low-resource speech recognition, Transformer, Conformer, domain adversarial training

1. Introduction

Recently, research on automatic speech recognition (ASR) for low-resource languages is getting more attention. We participate in the Open Automatic Speech Recognition Challenge 2020 (OpenASR20), which aims to assess the ASR technologies for low-resource languages [1]. The challenge offers ten low-resource languages and two different training conditions, Constrained and Unconstrained. In the Constrained training condition, we are only allowed to use the speech data provided by the challenge. In the Unconstrained training condition, any extra speech data can be used for training. The languages we participate in are Cantonese and Vietnamese, and the training condition we choose is the Constrained training condition.

In addition to OpenASR20, we participate in Formosa Speech Recognition Challenge 2020 (FSR-2020) [2], which focuses on a low-resource language, Taiwanese (Taiwanese Hokkien). There are three different tasks in FRS-2020. The first is to build a speech-to-text translation system that translates from Taiwanese speech to Traditional Chinese characters. The second and the third are to build Taiwanese speech recognizers that could output Taiwanese Southern Min Recommended Characters (Hàn-jī) or Taiwan Minnanyu Luomazi Pinyin (Tâi-lô). We participate in the second task and the third task of FSR-2020.

In this paper, we study Transformer-based end-to-end (E2E) ASR and Deep Neural Network Hidden Markov Model (DNN-HMM) ASR for the challenges. Transformer is a sequence-to-sequence architecture that was originally proposed for neural machine translation, which can replace recurrent neural networks (RNN) [3]. Recent researches show that Transformer can also be used for E2E ASR and DNN-HMM ASR [4, 5, 6, 7]. Furthermore, the encoder architecture named Conformer combines convolution neural networks (CNN) and Transformer to obtain both local and global context can achieve significant improvements in many ASR tasks [8, 9]. We also investigate the usage of domain adversarial training [10] that can reduce the mismatch between training and testing data. [11] shows that adversarial training is also helpful in the ASR task.

Time-Restricted Self-Attention (TRSA) [12] is a self-attention layer but only computes a limited number of frames to the left and right. TRSA can replace a Time Delay Neural Networks (TDNN) or Long Short-Term Memory (LSTM) layer in a DNN-HMM acoustic model trained with lattice-free MMI (maximum mutual information) [13, 14]. We try to build a Transformer-based acoustic model with TRSA and combine it with Factorized Time Delay Neural Networks (TDNN-F) [15].

The rest of this paper is organized as follows. Section 2 presents the E2E model and DNN-HMM acoustic model we propose. In Section 3, we explain our dataset and experimental setup of OpenASR20 and FSR-2020. Section 4 describes the experiments and results of the challenges. We present the conclusions in Section 5.

2. Model Architecture

2.1. End-to-End Model

Our E2E ASR model consists of a Conformer encoder and a Transformer decoder. A speaker classifier with a gradient reversal layer is employed after the encoder for domain adversarial training. The model architecture is illustrated in Figure 1.

2.1.1. Encoder Architecture

The input acoustic feature is a sequence of 80-dim FBank with 3-dim pitch. First, we subsample the input feature by two-layer CNN with ReLU activation, stride size of 2, kernel size of 3, and d_c channels, where d_c is the number of attention feature dimensions. Then we use the encoder named Conformer [8], which combines CNNs and Transformer modules [3] to capture both global and local information of input sequence. The architecture of the Conformer encoder is shown in the middle part of Figure 1.

2.1.2. Decoder Architecture

Transformer's decoder receives the output of Conformer encoder X_e and prefix sequence of token IDs $Y[1 : u] = Y[1], ..., Y[u]$. The architecture of the decoder is shown in the right part of Figure 1. Given the token IDs $Y[1 : u]$ and the output of encoder X_e, posterior probabilities of the output se-

http://dx.doi.org/10.21437/Interspeech.2021-358

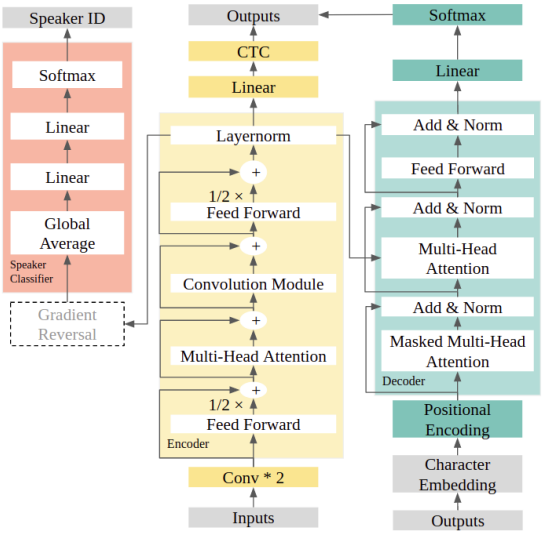

Figure 1: *Our E2E system model architecture. The encoded feature output by the Conformer encoder is received by the Transformer decoder and the connectionist temporal classification (CTC) module. A speaker classifier with a gradient reversal layer is employed for domain adversarial training to improve the robustness of the encoder.*

quence $p_{\text{s2s}}(Y|X_e)$ is calculated as:

$$[p_{\text{s2s}}(Y[2]|Y[1], X_e), ..., p_{\text{s2s}}(Y[u+1]|Y[1:u], X_e)] = \text{softmax}(Z_d W_{\text{att}} + b_{\text{att}}) \quad (1)$$

$$p_{\text{s2s}}(Y|X_e) = \prod_u p_{\text{s2s}}(Y[u+1]|Y[1:u], X_e) \quad (2)$$

where Z_d is the output of the decoder, $W_{\text{att}} \in \mathbb{R}^{d_c \times d_{\text{voc}}}$, $b_{\text{att}} \in \mathbb{R}^{d_{\text{voc}}}$ are learnable parameters and d_{voc} is the vocabulary size.

2.1.3. Training and Decoding

In the training stage, the loss function combines the negative log probability from the decoder and connectionist temporal classification (CTC) [16] for faster convergence [6]. The multi-task loss function is calculated as:

$$L_{\text{mtl}} = -\alpha \log p_{\text{s2s}}(Y|X_e) - (1-\alpha) \log p_{\text{ctc}}(Y|X_e) \quad (3)$$

where p_{ctc} is posterior probabilities predicted by the CTC module, and α is a hyperparameter. During decoding, we compute the sum of log probabilities from the Transformer decoder, CTC, and language model (LM):

$$\hat{Y} = \arg\max_{Y \in y^*} \{\lambda \log p_{\text{s2s}}(Y|X_e) + (1-\lambda) \log p_{\text{ctc}}(Y|X_e) + \gamma \log p_{\text{lm}}(Y)\} \quad (4)$$

where $p_{\text{lm}}(Y)$ denotes the LM probability of Y, λ and γ are hyperparameters, y^* is a set of output hypotheses.

2.1.4. Adversarial Training

Speakers' variety between the training set and the testing set may lead to poor recognition results. To address this problem, we employ adversarial training to encourage the encoder to learn speaker-invariant representation. A speaker classifier, which consists of two linear layers, receives the output of the encoder. We insert a gradient reversal layer [10] before the speaker classifier for domain adversarial training. The architecture of the speaker classifier is shown in the left part of Figure 1. The E2E model parameters are updated as:

$$\theta_{\text{dec}} \leftarrow \theta_{\text{dec}} - \epsilon \frac{\partial L_{\text{mtl}}}{\partial \theta_{\text{dec}}} \quad (5)$$

$$\theta_{\text{spk}} \leftarrow \theta_{\text{spk}} - \epsilon\beta \frac{\partial L_{\text{spk}}}{\partial \theta_{\text{spk}}} \quad (6)$$

$$\theta_{\text{enc}} \leftarrow \theta_{\text{enc}} - \epsilon\left(\frac{\partial L_{\text{mtl}}}{\partial \theta_{\text{enc}}} - \beta \frac{\partial L_{\text{spk}}}{\partial \theta_{\text{enc}}}\right) \quad (7)$$

where θ_{enc}, θ_{dec}, θ_{spk} refer to the parameters of encoder, decoder, and speaker classifier, respectively. L_{spk} is a cross-entropy loss function for the speaker classifier, and β is a hyperparameter.

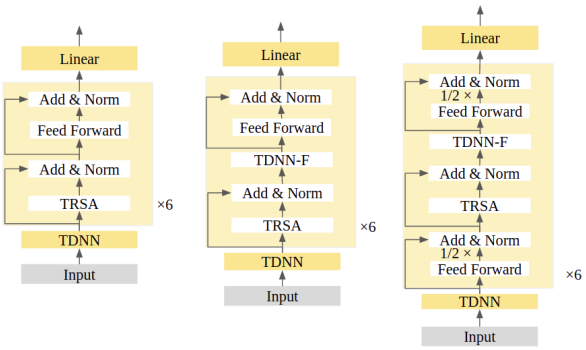

Figure 2: *Different acoustic models for our DNN-HMM systems. From left to right are TRSA-Transformer, TRSA-Transformer+TDNN-F, and TRSA-Transformer+TDNN-F+Macaron-FFN.*

2.2. Acoustic Model

Besides the E2E model, we implemented a DNN-HMM system for the Tâi-lô task of FSR-2020. The acoustic model of DNN-HMM starts with TDNN, followed by Transformer-linked architecture where the self-attention layer is replaced by a TRSA layer. Furthermore, we integrate the acoustic model with the TDNN-F layer and the Macaron-style structure with two half-step feed-forward networks (FFNs) [17]. Three different architecture of acoustic models is shown in Figure 2.

2.2.1. TRSA-Transformer Architecture

The input acoustic feature is 40-dim MFCCs with 3-dim pitch plus a 100-dim i-vector. The acoustic model starts with a TDNN layer with an output channel of 816, kernel size of 3, dilation rate of 3. Six layers of Transformer-linked architecture we refer to TRSA-Transformer are employed after the TDNN layer. Similar to Transformer, TRSA-Transformer consists of multi-head attention layers and FFNs, but the self-attention mechanism is replaced by Time-Restricted Self-Attention (TRSA) [12]. TRSA is the same as the original multi-head attention in [3] but focuses on local context rather than global context. TRSA layer is followed by the FFN, where the inner dimension of the FFN is 1024. Batch normalization and residual connection are employed after the FFN. The final layer of the acoustic model is a linear bottleneck of 256 with semi-orthogonal constraint, which is helpful in many circumstances [15].

2.2.2. Improving TRSA-Transformer Architecture

TDNN-F is factored form of TDNN with one of the two factors constrained to be semi-orthogonal. TDNN-F can effectively reduce the number of parameters while maintaining modeling power and the stability of training [15]. We insert the TDNN-F layer after the TRSA layer to further improve the performance of the acoustic model. The architecture of TRSA-Transformer+TDNN-F is shown in the middle part of Figure 2.

Inspired by [8, 17], we implement the Macaron-style structure with two half-step FFNs, one before the TRSA layer and the other after the TDNN-F layer. This architecture is more effective than a single FNN of the same number of parameters. The architecture of TRSA-Transformer+TDNN-F+Macaron-FNN is shown in the right part of Figure 2.

2.2.3. Training and Decoding

We use the lexicon provided by FSR-2020, which consists of Tâi-lô syllable to initials, finals, and tones. Six Hidden Markov Model - Gaussian Mixture Models (HMM-GMM) are trained for the states labels. The DNN acoustic model is trained with lattice-free maximum mutual information (LF-MMI) [13]. During decoding, we utilize a syllable-level tri-gram LM trained with the transcriptions of the training dataset.

3. Experiments

3.1. Dataset

OpenASR20 provides the Build dataset for training and the Dev dataset for validation, where the audio data consists of conversational telephone speech. The speech duration is around 10 hours for the Build dataset and 10 hours for the Dev dataset. We tokenize the transcriptions of the Cantonese dataset with characters, where the number of characters is 2053, including whitespace for the computation of the word error rate (WER). The Vietnamese dataset is tokenized with byte-pair-encoding (BPE) [18] using the open toolkit SentencePiece [19], where the number of BPE tokens is 3364. Whitespace is treated as a basic symbol while generating the BPE subword units.

TAT-Vol1-train, TAT-Vol1-eval, and PTS_TW-train are datasets provided from FSR-2020. The audio data of TAT-Vol1-train and TAT-Vol1-eval consist of daily conversations and short articles, and the PTS_TW-train dataset includes text and voice materials of TV shows from Taiwan Public Television Service Foundation (PTS). Table 1 shows the information of the datasets released by the challenge. We tokenize the Hàn-jī transcriptions with characters, where the number of characters is 4106. The Tâi-lô transcriptions are tokenized with characters, syllables, and BPE, where the numbers of the modeling units are 37, 2212, and 987, respectively. We noticed that some transcriptions of the PTS_TW-train dataset are not correctly aligned to the speech segment, and some audios contain non-Twaiwanese speech. To address this problem, we use CTC-Segmentation [20] to generate new alignments between the transcription and audio data. Furthermore, we use the log probability of the alignments as the threshold. PTS-1.5 denotes the alignments of the PTS_TW-train dataset with log probability greater than -1.5, and PTS-5.0 indicates which greater than -5.0.

We apply speed perturbation [21] and SpecAugment [22] for data augmentation.

Table 1: *System of transcriptions, number of utterances, and total speech duration for FSR-2020 datasets.*

Dataset	Transcriptions	Utterances	Hours
TAT-Vol1-train	Hàn-jī, Tâi-lô	23,104	41.76
TAT-Vol1-eval	Hàn-jī, Tâi-lô	2,664	4.78
PTS_TW-train	Hàn-jī	95	85.39
PTS-1.5	Hàn-jī	7,665	16.92
PTS-5.0	Hàn-jī	23,363	52.71

Table 2: *The comparison of different E2E models and their performance on the Dev dataset of OpenASR20.*

	Small	Medium	Large
Num Params (M)	6.5	20.5	44.7
Encoder Layers	6	8	12
Decoder Layers	3	4	6
Attention Dim	128	256	256
FFN Dim	1024	1024	2048
Cantonese WER(%)	63.0	63.6	**61.0**
Vietnamese WER(%)	**68.9**	71.5	70.6

3.2. Experimental Setups

3.2.1. End-to-End Model

The kernel size of the depthwise convolution in the Conformer is 32. The number of multi-heads is 4. The inner dimension of the speaker classifier is 256. The multi-task loss weight is $\alpha = 0.3$. We use the Adam optimizer [23] with square root learning rate scheduling [3]. The decoding hyperparameters λ and γ are 0.5 and 0.7. We use a 2-layer LSTM LM with 1024 neurons per-layer trained with the transcripts of the training dataset. The modeling unit used for the LM is the same as the E2E system that the LM integrates with. We implement our E2E system on the open ASR toolkit ESPnet [24].

3.2.2. Acoustic Model

The limitation of left and right context of TRSA is 5 and 2 in TRSA-Transformer. The dimension of the input vectors $\mathbf{q}, \mathbf{k}, \mathbf{v}$ for the TRSA layer are set to $d_q = d_k = 40$ and $d_v = 60$. The number of multi-heads is 12. The bottleneck dimension for the TDNN-F is 160, and the output channel dimension is 816. We implement our DNN-HMM system on the open ASR toolkit Kaldi [25].

4. Results

4.1. OpenASR20

The primary metric of OpenASR20 is WER. First, we explore three different E2E models, Small, Medium, and Large, using different combinations of Conformer encoder layers, decoder block layers, attention feature dimension, and FFN inner dimension. Table 2 describes their architecture hyperparameters and the result of different model sizes tested on the Dev dataset. We can see that the Large model has the best performance in Cantonese and the Small model has the best performance in Vietnamese.

In Table 3, we compare the result of different hyperparameter β on the Dev dataset for Cantonese with the Large E2E

Table 3: *The comparison of different hyperparameter β on the Dev dataset for Cantonese with the Large E2E model.*

β	0	0.3	0.5	0.7	1
WER(%)	62.9	62.1	61.0	**60.7**	61.8

Table 4: *The comparison of different E2E models and their performance on the TAT-Vol1-eval dataset for the Hàn-jī task of FSR-2020.*

Encoders	Transformer	Conformer		
Num Params (M)	29.5	9.9	29.6	45.4
Encoder Layers	12	10	12	12
Decoder Layers	6	4	6	6
Attention Dim	256	128	256	256
FFN Dim	2048	1024	1024	2048
CER(%)	36.3	27.1	**26.6**	28.7

model, where β controls the proportion of adversarial learning in the loss function. Compared with not using adversarial training, when β is 0.7, WER can be reduced from 62.9% to 60.7%.

The best result of the OpenASR evaluation is WER of 61.45% for Cantonese, which is ranked 4th out of 6 teams. The WER of 74.61% for Vietnamese is ranked 3rd out of 4 teams [26]. Although adversarial training is helpful for the performance of the E2E model, there's still a gap compared to other teams. We will compare these results with the FSR-2020 final evaluation results in section 4.2.

4.2. FSR-2020 Hàn-jī Task

The primary metric of the FSR-2020 Hàn-jī task is character error rate (CER). We compare different E2E models on the TAT-Vol1-eval dataset and the performance between the Transformer encoder and Conformer encoder when the model size is similar. Table 4 shows that when the number of parameters is close, the Conformer encoder achieves a better result than the Transformer encoder.

Next, we compare the result of different training datasets for the E2E model. When we add the PTS-1.5 dataset to training data with the TAT-Vol1-train dataset, we can improve our CER from 26.6% to 21.2%. When we use the PTS-5.0 dataset and the TAT-Vol1-train dataset as training data, we can further improve CER to 19.1%.

The E2E model trained with the PTS-1.5 dataset and the TAT-Vol1-train dataset achieved 48.0% CER in the final evaluation of the FSR-2020 Hàn-jī task, which is ranked 8th out of 21 results. And the model trained with the PTS-5.0 dataset and the TAT-Vol1-train dataset achieved 43.4% CER, which is ranked 4th out of 21 results. Except for the model size and the unused adversarial training, this system is the same as in OpenASR20. However, we utilize CTC-Segmentation to obtain a better quality of additional training data for the Hàn-jī task, which allows us to outperform most of the other teams.

4.3. FSR-2020 Tâi-lô Task

The primary metric of the FSR-2020 Tâi-lô task is syllable error rate (SER). We compare the performance between the Transformer encoder and Conformer encoder of the E2E model on the TAT-Vol1-eval dataset. We also compare different ways to tokenize the transcriptions of the TAT-Vol1-train dataset. In Table 5, we observe that the Transformer encoder shows better results than the Conformer encoder. The way we tokenize the transcriptions is also significant for performance.

Table 5: *The comparison of different E2E encoders and token types on the TAT-Vol1-eval dataset for the Tâi-lô task of FSR-2020.*

Encoders	Token Type	SER(%)
Transformer	Character	20.8
	Syllable	20.9
	BPE	**18.3**
Conformer	Character	-
	Syllable	23.6
	BPE	19.3

Table 6: *The comparison of different acoustic models in the DNN-HMM system on the TAT-Vol1-eval dataset for the FSR-2020 Tâi-lô task.*

Acoustic Models	SER(%)
TRSA-Transformer	16.3
TRSA-Transformer+TDNN-F	15.4
TRSA-Transformer+TDNN-F+Macaron-FNN	**15.3**

The comparison of different acoustic models in the DNN-HMM system on the TAT-Vol1-eval dataset for the FSR-2020 Tâi-lô task is shown in Table 6. We can see that TDNN-F improves the performance of TRSA-Transformer while having a Macaron-style FFN is also helpful.

The DNN-HMM system achieved 27.1% SER in the final evaluation of the FSR-2020 Tâi-lô task, which is ranked 16th out of 21 results. The E2E model achieved 25.4% SER, which is ranked 15th out of 21 results. Although our DNN-HMM system performs better than our E2E model on the TAT-Vol1-eval dataset, the E2E model achieved a better result in the final test.

5. Conclusion

In participating in OpenASR20 and FSR-2020, we propose an E2E ASR system with the Conformer encoder and Transformer decoder. Besides, we employ a speaker classifier for adversarial training. In OpenASR20, we have achieved 61.45% WER for Cantonese and 74.61% WER for Vietnamese with the proposed methods. In addition to the E2E system, we implement a DNN-HMM ASR system where the acoustic model consists of Transformer-linked architecture augmented with TDNN-F. In FSR-2020, we have achieved 43.4% CER for the Hàn-jī task and 25.4% SER for the Tâi-lô task.

In the future, we want to investigate how to further improve our system performance in the constrained training data. For instance, though the audio data is limited, additional text data might further improve our LM. It is also important to find out why our DNN-HMM system outperforms the E2E system in the Tâi-lô task validation set but got a worse performance than the E2E system in the final results.

6. References

[1] Openasr Challenge — NIST. [Online]. Available: https://www.nist.gov/itl/iad/mig/openasr-challenge

[2] Y.-F. Liao, C.-Y. Chang, H.-K. Tiun, H.-L. Su, H.-L. Khoo, J. S. Tsay, L.-K. Tan, P. Kang, T.-g. Thiann, U.-G. Iunn *et al.*, "Formosa speech recognition challenge 2020 and taiwanese across taiwan corpus," in *2020 23rd Conference of the Oriental COCOSDA International Committee for the Co-ordination and Standardisation of Speech Databases and Assessment Techniques (O-COCOSDA)*. IEEE, 2020, pp. 65–70.

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[4] L. Dong, S. Xu, and B. Xu, "Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5884–5888.

[5] Y. Wang, A. Mohamed, D. Le, C. Liu, A. Xiao, J. Mahadeokar, H. Huang, A. Tjandra, X. Zhang, F. Zhang *et al.*, "Transformer-based acoustic modeling for hybrid speech recognition," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6874–6878.

[6] S. Karita, N. E. Y. Soplin, S. Watanabe, M. Delcroix, A. Ogawa, and T. Nakatani, "Improving Transformer-Based End-to-End Speech Recognition with Connectionist Temporal Classification and Language Model Integration," in *Proc. Interspeech 2019*, 2019, pp. 1408–1412. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-1938

[7] S. Karita, N. Chen, T. Hayashi, T. Hori, H. Inaguma, Z. Jiang, M. Someki, N. E. Y. Soplin, R. Yamamoto, X. Wang *et al.*, "A comparative study on transformer vs rnn in speech applications," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 449–456.

[8] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, and R. Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Proc. Interspeech 2020*, 2020, pp. 5036–5040. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2020-3015

[9] P. Guo, F. Boyer, X. Chang, T. Hayashi, Y. Higuchi, H. Inaguma, N. Kamo, C. Li, D. Garcia-Romero, J. Shi, J. Shi, S. Watanabe, K. Wei, W. Zhang, and Y. Zhang, "Recent developments on espnet toolkit boosted by conformer," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 5874–5878.

[10] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[11] Y. Shinohara, "Adversarial multi-task learning of deep neural networks for robust speech recognition." in *Interspeech*. San Francisco, CA, USA, 2016, pp. 2369–2372.

[12] D. Povey, H. Hadian, P. Ghahremani, K. Li, and S. Khudanpur, "A time-restricted self-attention layer for asr," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5874–5878.

[13] D. Povey, V. Peddinti, D. Galvez, P. Ghahremani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi." in *Interspeech*, 2016, pp. 2751–2755.

[14] V. Peddinti, Y. Wang, D. Povey, and S. Khudanpur, "Low latency acoustic modeling using temporal convolution and lstms," *IEEE Signal Processing Letters*, vol. 25, no. 3, pp. 373–377, 2017.

[15] D. Povey, G. Cheng, Y. Wang, K. Li, H. Xu, M. Yarmohammadi, and S. Khudanpur, "Semi-orthogonal low-rank matrix factorization for deep neural networks." in *Interspeech*, 2018, pp. 3743–3747.

[16] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*, 2006, pp. 369–376.

[17] Y. Lu*, Z. Li*, D. He, Z. Sun, B. Dong, T. Qin, L. Wang, and T. yan Liu, "Understanding and improving transformer from a multi-particle dynamic system point of view." in *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*, 2020. [Online]. Available: https://openreview.net/forum?id=pxlqJa21C

[18] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 1715–1725. [Online]. Available: https://www.aclweb.org/anthology/P16-1162

[19] T. Kudo and J. Richardson, "SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 66–71. [Online]. Available: https://www.aclweb.org/anthology/D18-2012

[20] L. Kürzinger, D. Winkelbauer, L. Li, T. Watzel, and G. Rigoll, "Ctc-segmentation of large corpora for german end-to-end speech recognition," in *International Conference on Speech and Computer*. Springer, 2020, pp. 267–278.

[21] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[22] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," in *Proc. Interspeech 2019*, 2019, pp. 2613–2617. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2680

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[24] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "ESPnet: End-to-end speech processing toolkit," in *Proceedings of Interspeech*, 2018, pp. 2207–2211. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1456

[25] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. CONF. IEEE Signal Processing Society, 2011.

[26] Openasr Challenge Results — NIST. [Online]. Available: https://www.nist.gov/itl/iad/mig/openasr20-challenge-results