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

An End-to-End Dialect Identification System with Transfer Learning from a Multilingual Automatic Speech Recognition Model

Ding Wang[1*], *Shuaishuai Ye*[1*], *Xinhui Hu*[1], *Sheng Li*[2], *and Xinkang Xu*[1]

[1]Hithink RoyalFlush AI Research Institute, Zhejiang, China
[2]National Institute of Information and Communications Technology (NICT), Kyoto, Japan

{wangding2,yeshuaishuai,huxinhui,xuxinkang}@myhexin.com
sheng.li@nict.jp

Abstract

In this paper, we propose an end-to-end (E2E) dialect identification system trained using transfer learning from a multilingual automatic speech recognition (ASR) model. This is also an extension of our submitted system to the Oriental Language Recognition Challenge 2020 (AP20-OLR). We verified its applicability using the dialect identification (DID) task of the AP20-OLR. First, we trained a robust conformer-based joint connectionist temporal classification (CTC) /attention multilingual E2E ASR model using the training corpora of eight languages, independent of the target dialects. Second, we initialized the E2E-based classifier with the ASR model's shared encoder using a transfer learning approach. Finally, we trained the classifier on the target dialect corpus. We obtained the final classifier by selecting the best model from the following: (1) the averaged model in term of the loss values; and (2) the averaged model in term of classification accuracy.

Our experiments on the DID test-set of the AP20-OLR demonstrated that significant identification improvements were achieved for three Chinese dialects. The performances of our system outperforms the winning team of the AP20-OLR, with the largest relative reductions of 19.5% in C_{avg} and 25.2% in EER.

Index Terms: dialect identification, end-to-end network, multilingual ASR, transfer learning

1. Introduction

Dialect identification (DID) refers to identifying dialect categories from utterances, and its task is to recognize a speaker's regional dialect, within a predetermined language, given the acoustic signal alone. DID is a special case of language identification (LID) in the general sense; however, it is a considerably challenging task because similarities between dialects tend to be higher than those used in the general LID task [1–4]. Spoken LID is usually the front end of many multilingual applications, such as speech translation, multilingual speech recognition, multilingual conversational systems, etc.

There have been many studies for LID or DID so far. Due to their characteristics and relationships with individual speakers, many techniques used for speaker recognition are also successfully adopted for LID. In [5], the authors optimized the combination between the Maximum A Posteriori (MAP) and the Maximum Likelihood Linear Regression (MLLR), both of which are practical approaches for speaker recognition), and developed an approach to detecting and utilizing the degree of accent in a Shanghainese accented ASR. With the development of deep neural networks (DNN) in speech applications, a significant step forward was obtained by combining i-vector or x-vector and DNNs for both speaker recognition and LID [6–8]. More recently, dialect and accent recognition have begun to receive attention from the speech science and technology communities [9–17]. [17] proposed a method that combined DNNs and Recurrent Neural Networks (RNNs) to be respectively trained on long-term statistic features and short-term acoustic features for DID. In [13], the authors proposed an E2E-based DID model for Arabic dialectal speech. They applied a convolutional neural network (CNN) and conducted extensive experiments in which they compared different acoustic features and data augmentation methods. [18] utilized high-order LID-senone statistics for two E2E DNN-CNN network variants, and conducted experiments on a set of 23 languages of NIST LRE 2009 corpora. With such architectures, their performances were significantly improved when they were compared to the state-of-the-art deep bottleneck features (DBF) /i-vector systems.

When the E2E approaches demonstrate their strengths in LID (or DID) performance, they also face data sparseness and training efficiency since they generally require many training data and a long training time. To deal with these problems, many strategies such as transfer learning [19–21], multi-task training [22, 23], etc., have been widely utilized in the machine learning fields. The basic approach of transfer learning is to initialize the encoder and/or prediction network of the target model with the pre-trained models. As CTC/attention E2E-based multilingual ASR model has shown its superiority to the other models [24], its shared encoder is regarded as having strong capabilities to discriminate languages, and can be expected to improve the DID task. To promote LID and cope with the real challenge existing in LID tasks, the oriental language recognition (OLR) has been held annually since 2016. This challenge has attracted dozens of teams from around the world [8, 25–28]. The RoyalFlush team has participated in this challenge since the AP19-OLR. This study is an extension of our submitted system for the AP20-OLR. We continue to follow the AP20-OLR instructions to have fair comparisons, including using the same data sets and evaluation metrics in experiments.

Motivated by the successes of transfer learning approaches in the DNN-based ASR systems, in this study, we propose the architecture of an E2E-based classifier for a DID task in which initialization is performed by the shared encoder of a multilingual E2E ASR model.

The contributions of this work are summarized as follows: We use a multilingual E2E ASR system to initialize our DID system. The languages used to build the ASR systems are independent of DID tasks' target languages. We prove that such an architecture is robust by comparing it with a system built using an individual language. We propose a conformer-based

*Equal contribution.

http://dx.doi.org/10.21437/Interspeech.2021-374

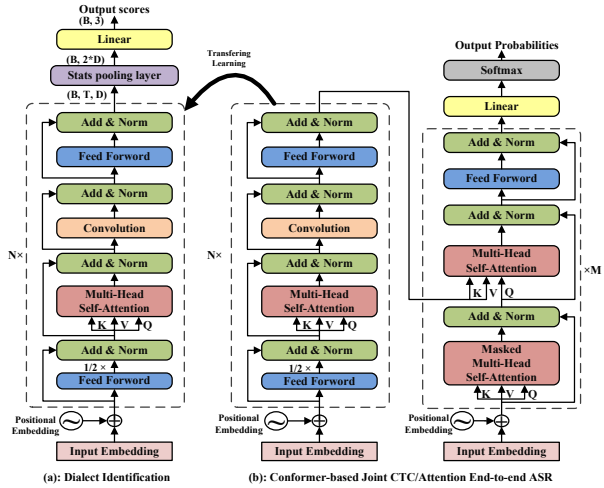

Figure 1: *Structure of our DID system. (B, T, D) represents the size set of the encoder's output, where B is the batch size, T is a time dimension, and D is the output dimension of self-attention.*

encoder to construct the DID classifier and verify that it is superior to the transformer-based model. Experiments conducted on a public corpus of the AP20-OLR demonstrated the effectiveness of our proposed methods, and the performances showed having surpassed the winning team of the challenge.

2. System configuration

Figure 1 shows our DID system configuration. Figure 1 (a) shows an E2E DID classifier composed of a conformer-encoder, statistics pooling layer, and linear layer. Figure 1 (b) shows a conformer [29] based joint CTC/attention multilingual ASR model trained on a set of corpora of several different languages. Considering the close correlativity between DID and speech recognition, we first train a conformer-based joint CTC/attention multilingual E2E ASR model on the training set *AP20-E2E-ASR-train*, which we explain later. We then train the DID classifier whose encoder is initialized by the above-mentioned pre-trained multilingual ASR model.

2.1. Multilingual E2E ASR model

Our E2E ASR model adopts a hybrid CTC/attention architecture [30] that consists of three components: a shared encoder, an attention decoder, and a CTC module. Depending on the type of the shared encoder structures, the E2E ASR model is divided into a transformer-based structure [31, 32] and a conformer-based structure [29]. The conformer-based E2E ASR model, that is, the convolution-augmented transformer, has been demonstrated to outperform the transformer in ASR tasks and presents a trend of replacing the transformer as a standard component in E2E ASR architectures [29]. Hence, we focus on the conformer-based ASR model in this paper. For comparison, we also explore the transformer-based ASR model. We establish the hybrid architecture of the ASR in a multi-task learning scheme. The training process is to jointly optimize the weight-sum of the loss of the attention decoder L_{att} and the CTC loss L_{ctc} [33] shown as:

$$L = \alpha L_{ctc} + (1-\alpha) L_{att} \quad (1)$$

where the hyper-parameter α represents the weight of the CTC loss. We chose α as 0.3 in this study based on the empirical values from ASR experiments. It has been shown that the introduction of the CTC module can help to ensure appropriate alignments and fast converge [32].

For multilingual ASR tasks, we adopt a language-independent architecture in the same manner as in [34] so that all the target languages can share the same network architecture and parameters. In addition, the multilingual ASR modeling unit is a single character for all languages, and some characters are common to all languages. All the target languages share the same vocabulary. The vocabulary size in our multilingual ASR model varies with the types of language. A sample vocabulary is shown as follows:

a, ..., z, A, ..., Z, 你, ..., 韵, あ, ..., う, Ǐ, ..., Ƙ, é, ..., ũ, ت, ..., ى

We train the E2E model using a multilingual corpus. We introduce detailed information about the training set in Section 3.

2.2. End-to-end DID classifier

As shown in Figure 1 (a), our DID classifier is also an E2E-based architecture. First, the encoder of the DID classifier embeds the input frame into an embedding vector. Second, the statistics pooling layer aggregates all T frame-level embedding vectors from the classifier's encoder and computes their mean and standard deviation. Finally, the statistics are concatenated into a 2*D dimensional vector and fed into a subsequent linear layer to obtain the final output scores. The process aggregates information across the time dimension so that subsequent linear layers operate on the entire utterance. To make full use of the linguistic information that exists in the ASR model, we first initialize its encoder by transferring the E2E ASR's encoder to the classifier and then using the target dialects' training data to fine-tune the whole classifier network.

The conformer-based shared encoder network consists of multi-head self-attention modules, convolution modules, feed-forward modules, and normalization modules. The shared encoder can learn the linguistic information to contribute more to distinguishing different dialects [34]. Instead of using cross-entropy loss, the L_{att} loss in Equation (1) is used to optimize the DID task's classification performance directly. To improve the system robustness, we further process all intermediate models saved during training the classifier from two aspects: (1) Ten models with the smallest loss values are selected and are averaged into a new model M_{loss}. (2) Ten models with the highest classification accuracy relative to a validation set are selected and are also averaged into a new model M_{acc1}. Finally, from the above two models, M_{loss} and M_{acc1}, that with the lowest average cost performance (C_{avg}) on the development set is selected as the final classifier model.

3. Experiments

3.1. Experimental data

3.1.1. Brief introduction to the DID task of the AP20-OLR

Task 2 of the AP20-OLR [28] is for DID, where three Chinese dialectual data sets, that is, Hokkien, Sichuanese, and Shanghainese, were provided for training and evaluation. In addition to the above mentioned three dialects in the test-set, three other non-target languages, that is, Catalan, Greek, and Telugu, were included to compose the open-set DID. We participated in this task, and achieved second place.

Table 1: *Data sets used in our systems*

Data sets	Set name in our paper	Data composition [28]
Training set	AP20-E2E-ASR-train	AP16-OL7, AP17-OL3, THCHS30
	AP20-OLR-dialect	AP20-OLR-dialect
Development set	AP20-OLR-dialect-dev	AP19-OLR-dev-task3 AP19-OLR-zero
test-set	AP20-OLR-dialect-test	AP20-OLR-dialect-test

3.1.2. Dataset

For comparison, all our experiments were conducted based on the requirements of task 2 of AP20-OLR. Therefore, the data used for our experiments was limited to those stated in the challenge. For convenience in the following description, we divide the entire data set into two categories:

(1) *AP20-E2E-ASR-train* refers to as the data set used for training the ASR system, and (2) *AP20-OLR-dialect* [28] refers to the data set used for training the DID classifier. Meanwhile, *AP20-OLR-dialect-dev* and *AP20-OLR-dialect-test* are used to refer to dev-set and test-set, respectively. Detailed information about the original sources of these data set is shown in Table 1.

There were ten languages (Cantonese, Mandarin, Indonesian, Japanese, Russian, Korean, Vietnamese, Kazakh, Tibetan, Uyghur) in the *AP20-E2E-ASR-train*. Here, it should be noted that the three target dialects were not included in this training data set. This means that the E2E ASR was independent of the target dialects. However, both dev-set and evaluation-set contained their own corresponding interference languages. Specifically, Catalan, Greek, and Telugu are contained in the dev-set, and Mandarin, Malay, and Thai were contained in the evaluation-set.

3.2. Evaluation metrics

Unlike the general dialect or accent identification studies in which the recognition accuracy of the individual target language is generally emphasized, the average cost performance C_{avg} [28] and the equal error rate (EER) were adopted as the evaluation metrics for the DID task in the challenge. We followed the challenge's guidance and adopted these two metrics to evaluate our DID systems. These two metrics evaluate a DID system's system performance from different perspectives, which offers an entire picture of the tested system's capability. For open-set testing conditions, all interfering languages will be seen as one unknown language in the computation of C_{avg}. Because C_{avg} has a sense of cost, the lower the value, the better the recognition. By contrast, the EER is a value on the receiver operating characteristic(ROC) curve where the probabilities of both false acceptance (P_{fa}) and false rejection (P_{miss}) are equal.

3.3. Systems for comparison

There is a comparable baseline system[1] provided by AP20-OLR, which uses an extended TDNN x-vector model with a logistic regression back-end classifier. Two versions of the baseline systems were implemented on Kaldi and PyTorch platforms. Before training, data augmentation, including speed and volume perturbation, was adopted to increase the amount and diversity of the training data. Specifically, the baselines used the i-vector or x-vector framework as the front-end linguistic feature extractor and used a back-end logistic regression classifier to identify dialects. Simultaneously, the feature normalization strategy was used to improve the performance of the classifier. The baseline systems' performances are shown in the Group1 of Table 2. In addition to the baseline systems from AP20-OLR, we also list the winning team results and our system submitted to the challenge on the Group2 of Table 2.

3.4. Experimental Settings

We conducted all experiments on the ESPnet platform but performed feature extraction on the Kaldi platform. All codes in our experiments were modified from a public code set[2]. The acoustic feature of 83 dimensions, 80-dimensional FBank + 3-dimensional pitch feature, was extracted from every speech frame with a frame-length of 25ms and a frame-shift of 10ms. To curb the over-fitting of the DID system, we applied several data augmentation methods, such as speed perturbation and spectral augmentation, in our systems. We trained the model for 40 epochs and 50 epochs for DID and ASR, respectively, using 4 NVIDIA 1080Ti GPUs with 11GB memory. We set the batch size according to the maximum memory of the GPU because of the different network architecture. More detailed hyperparameters and implementations can be acquired in our code [3].

Because Korean and Kazakh's characters were garbled in our development environment, these two languages were not used for building the multilingual ASR model. We only trained the multilingual ASR model, which was used for the DID classifier's initialization, on eight languages (see Table 2).

3.5. Experimental results and analysis

All of our experiments were conducted following five aspects as research objectives: (1) how the performance of our E2E DID changed compared with traditional methods; (2) the influence of transfer learning from multilingual ASR on the performance of DID; (3) the influence of the language number in the multilingual ASR that was used to initialize the DID classifier; (4) the influence of different network architectures and their hyperparameters, such as network depth; and (5) the influence of the main language to which the dialects belong. The experimental results are shown in Table 2. From this table, we can compare the performance of models with different settings.

3.5.1. E2E system vs x-vector/i-vector system

By comparing Group1 with the transformer and conformer-based models, we can see that the performances of our two types of E2E models were generally better than those of the baseline models, which are based on the traditional i-vector and x-vector. These facts can be attributed to the ability of the E2E model to extract global features from long sequences. By contrast, if the CTC-based model was not used as an auxiliary component, the E2E model was easily prone to over-fitting and could not obtain satisfactory results on the test-sets. We found final performance to be related to the layer depth of the classifier encoder. As shown in Group3 and Group4 in the results table, the classifier's performance improved first as the layer number increased, but the performance then declined when the layer number exceeded a certain value. Additionally, many transformer-based and conformer-based systems outperformed the winning team's system and the system that we submitted to the challenge in which a transformer with 12 encoder layers and only 6 languages for training the ASR were used.

[1] https://github.com/Snowdar/asv-subtools/tree/master/recipe/ap-olr2020-baseline

[2] https://github.com/R1ckShi/AESRC2020/tree/master

[3] https://github.com/YeBobr/End-to-end-dialect-identification

Table 2: *DID results for systems with different settings on dev-set and test-set. Languages[x] denotes the language IDs for training the ASR model, and the number x refers to the language number. Here, the 'L_1-L_8' represent Cantonese, Mandarin, Indonesian, Japanese, Russian, Vietnamese, Tibetan, Uyghur, respectively.*

Group	System	Encoder layers	ASR-init	Languages[x]	Dev-sets		Test-sets	
					Cavg	EER%	Cavg	EER%
Group1	Baseline i-vector[Kaldi]	-	No	-	0.0703	9.33	0.2214	23.94
	Baseline x-vector[Kaldi]	-	No	-	0.0807	14.67	0.2117	22.25
	Baseline x-vector[Pytorch]	-	No	-	0.0849	12.40	0.1752	19.74
Group2	Winning team	-	No	-	-	-	0.0738	11.97
	Our submitted transformer	12	Yes	L_1-L_5,L_8[6]	0.0351	6.34	0.0871	11.97
Group3	Transformer	12	No	-	0.0553	10.17	0.1570	18.97
	Transformer	18	No	-	0.0519	8.38	0.1407	17.15
	Transformer	24	No	-	0.0510	8.12	0.1442	17.49
Group4	Transformer	12	Yes	L_1-L_8[8]	0.0280	4.75	0.0777	10.97
	Transformer	18	Yes	L_1-L_8[8]	0.0214	4.09	0.0672	9.34
	Transformer	24	Yes	L_1-L_8[8]	0.0317	5.28	0.0794	10.93
Group5	Conformer	18	Yes	L_1,L_2[2]	0.0279	4.62	0.0724	10.88
Group6	Conformer	18	Yes	L_1-L_5,L_8[6]	0.0300	4.75	0.0678	10.45
	Conformer	18	Yes	L_1-L_4,L_6,L_7[6]	0.0267	4.29	0.0659	9.93
	Conformer	18	Yes	L_3-L_8[6]	0.0443	12.21	0.1279	17.35
	Conformer	18	Yes	L_1-L_2,L_4-L_8[6]	0.0276	3.50	0.0619	10.54
Group7	Conformer	18	Yes	L_1-L_8[8]	0.0203	3.10	0.0594	8.95

3.5.2. With ASR vs. Without ASR for initialization of the classifier

Comparing Group4 with Group3 (for the former, an ASR was used, and for the latter, no ASR was used), we can see that Group4 was distinctly superior to Group3. This means that the ASR's initialization greatly contributed to these improvements. For example, using a transformer system with an encoder of 18 layers, the C_{avg} was decreased from 0.1407 (without ASR-based initialization) to 0.0672 (with ASR-based initialization).

3.5.3. Conformer vs Transformer

By comparing Group7 with Group4, we can see that the conformer-based E2E outperformed the transform-based E2E. With respect to the test-set, the C_{avg} reduced from 0.0672 (in transformer) to 0.0594 (in conformer). Meanwhile, the EER% reduced from 9.343 (in transformer) to 8.95 (in conformer).

3.5.4. Influence of number of training languages for multilingual ASR

We conducted several groups (Group5 and Group6) of comparative experiments to verify the language numbers' influence on the multilingual ASR, which we used for the classifier's initialization. We found that the classifier's performance improved as the number of training languages increased. This fact can be explained that more acoustic characteristics embedded in different languages were learned as the training languages increased and the acoustical coverage was extended. So the DID task was benefited from this extension.

3.5.5. Influence of the main language

It is reasonable to consider that the primary language has a strong influence on its dialects. To verify this, we conducted experiments to investigate the main language's role to which the dialect belonged in the DID tasks. Although Cantonese itself is a Chinese dialect, it is spoken by many people in southern China and overseas, and it has great influences on many other dialects due to regional relationships. Same as the other dialects, Cantonese shares the same writing system based on Chinese characters. So, we considered Mandarin (L1:zh-cn) and Cantonese (L2:ct-zh) as the two main languages of our tasks' dialects. We conducted two extreme experiments, one with only the two main languages (Group5) and the other with no main languages at all (Group6 with L3-L8[6]). The results demonstrated that the main language's influence on dialect recognition was far greater than that of other languages. However, by comparing the results of experiments in Group6, we found that other languages also enhanced system performance.

4. Conclusions

In this paper, we proposed an E2E DID system with transfer learning from a multilingual E2E ASR model. To improve the robustness of the DID system, we customized ten intermediate models from two aspects, loss and accuracy, to obtain the final classifier. With the transfer learning scheme and model customization, our system achieved satisfactory results for the DID task of AP20-OLR. Extensive analysis was made on the optimal architecture, hyper parameters, and selection of training data. Our system's performance outperformed the winning system of the AP20-OLR. Compared with the winning team's result, our DID system's C_{avg} decreased by 19.5% relatively (from 0.0783 to 0.0594), and the $EER\%$ decreased by 25.2% relatively (from 11.97 to 8.95). The experiments demonstrated that the initialization from a pre-trained E2E ASR greatly improved the DID performance on task 2 of the AP20-OLR, with a relative reduction of 52.23% for the C_{avg} (from 0.1407 to 0.0672). Our future work is summarized: (1) A more detailed quantitative analysis of the main language's influence on the DID task is a topic. (2) We will verify the effectiveness of the proposed approach on other dialect or language identification tasks and compare the proposed approach with other approaches with bottleneck features.

5. References

[1] A. Ali, N. Dehak, P. Cardinal, S. Khurana, S. H. Yella, J. Glass, P. Bell, and S. Renals, "Automatic dialect detection in arabic broadcast speech," *arXiv preprint arXiv:1509.06928*, 2015.

[2] M. Najafian, S. Khurana, S. Shan, A. Ali, and J. Glass, "Exploiting convolutional neural networks for phonotactic based dialect identification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5174–5178.

[3] M. Zampieri, S. Malmasi, P. Nakov, A. Ali, S. Shon, J. Glass, Y. Scherrer, T. Samardžić, N. Ljubešić, J. Tiedemann *et al.*, "Language identification and morphosyntactic tagging: The second VarDial evaluation campaign." in *Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018)*, 2018.

[4] S. Shon, W.-N. Hsu, and J. Glass, "Unsupervised representation learning of speech for dialect identification," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 105–111.

[5] Y. Zheng, R. Sproat, L. Gu, I. Shafran, H. Zhou, Y. Su, D. Jurafsky, R. Starr, and S.-Y. Yoon, "Accent detection and speech recognition for shanghai-accented mandarin," in *Ninth European Conference on Speech Communication and Technology*, 2005.

[6] P. Cardinal, N. Dehak, Y. Zhang, and J. Glass, "Speaker adaptation using the i-vector technique for bottleneck features," in *Interspeech*, 2015.

[7] N. Dehak, P. A. Torres-Carrasquillo, D. Reynolds, and R. Dehak, "Language recognition via i-vectors and dimensionality reduction," in *Interspeech*, 2011.

[8] Z. Li, M. Zhao, J. Li, Y. Zhi, L. Li, and Q. Hong, "The xmuspeech system for the ap19-olr challenge," *Proc. Interspeech 2020*, pp. 452–456, 2020.

[9] F. Biadsy, "Automatic dialect and accent recognition and its application to speech recognition," Ph.D. dissertation, Columbia University, 2011.

[10] H. Behravan, V. Hautamäki, S. M. Siniscalchi, T. Kinnunen, and C.-H. Lee, "I-vector modeling of speech attributes for automatic foreign accent recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 1, pp. 29–41, 2015.

[11] K. Mannepalli, P. N. Sastry, and M. Suman, "Mfcc-gmm based accent recognition system for telugu speech signals," *International Journal of Speech Technology*, vol. 19, no. 1, pp. 87–93, 2016.

[12] H. Leach, K. Watson, and K. Gnevsheva, "Perceptual dialectology in northern england: Accent recognition, geographical proximity and cultural prominence," *Journal of Sociolinguistics*, vol. 20, no. 2, pp. 192–211, 2016.

[13] S. Shon, A. Ali, and J. Glass, "Convolutional neural networks and language embeddings for end-to-end dialect recognition," *arXiv preprint arXiv:1803.04567*, 2018.

[14] 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 *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2018, pp. 4749–4753.

[15] W. Wang, C. Zhang, and X. Wu, "Sar-net: A end-to-end deep speech accent recognition network," in *arXiv preprint arXiv:2011.12461*, 2020.

[16] W. Hou, Y. Dong, B. Zhuang, L. Yang, J. Shi, and T. Shinozaki, "Large-Scale End-to-End Multilingual Speech Recognition and Language Identification with Multi-Task Learning," in *Proc. Interspeech 2020*, 2020, pp. 1037–1041.

[17] Y. Jiao, M. Tu, V. Berisha, and J. M. Liss, "Accent identification by combining deep neural networks and recurrent neural networks trained on long and short term features." in *Interspeech*, 2016, pp. 2388–2392.

[18] M. Jin, Y. Song, I. McLoughlin, W. Guo, and L.-R. Dai, "End-to-end language identification using high-order utterance representation with bilinear pooling," in *Proc. Interspeech 2017*, 2017.

[19] H. Yan, Y. Dong, C. Liu, and Y. Gong, "Multi-accent deep neural network acoustic model with accent-specific top layer using the kld-regularized model adaptation," *Interspeech*, 2014.

[20] W. Chan, N. R. Ke, and I. Lane, "Transferring knowledge from a RNN to a DNN," *Computer Science*, 2015.

[21] J. Kunze, L. Kirsch, I. Kurenkov, A. Krug, J. Johannsmeier, and S. Stober, "Transfer learning for speech recognition on a budget," *arXiv preprint arXiv:1706.00290*, 2017.

[22] D. Yu and L. Deng, "Efficient and effective algorithms for training single-hidden-layer neural networks," *Pattern recognition letters*, vol. 33, no. 5, pp. 554–558, 2012.

[23] S. Zhou, S. Xu, and B. Xu, "Multilingual end-to-end speech recognition with a single transformer on low-resource languages," in *arXiv eprint arXiv:1806.05059*, 2018.

[24] M. Karafiát, M. K. Baskar, S. Watanabe, T. Hori, M. Wiesner, and J. Černocký, "Analysis of Multilingual Sequence-to-Sequence Speech Recognition Systems," in *Proc. Interspeech 2019*, 2019.

[25] Z. Tang, D. Wang, Y. Chen, and Q. Chen, "Ap17-olr challenge: Data, plan, and baseline," in *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2017.

[26] Z. Tang, D. Wang, and Q. Chen, "Ap18-olr challenge: Three tasks and their baselines," in *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2018.

[27] Z. Tang, D. Wang, and L. Song, "Ap19-olr challenge: Three tasks and their baselines," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2019.

[28] Z. Li, M. Zhao, Q. Hong, L. Li, Z. Tang, D. Wang, L. Song, and C. Yang, "Ap20-olr challenge: Three tasks and their baselines," in *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2020, pp. 550–555.

[29] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[30] 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.

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

[32] 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.

[33] 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.

[34] S. Watanabe, T. Hori, and J. R. Hershey, "Language independent end-to-end architecture for joint language identification and speech recognition," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2017, pp. 265–271.