E2E-based Multi-task Learning Approach to Joint Speech and Accent Recognition

Jicheng Zhang[1], Yizhou Peng[1], Van Tung Pham[2], Haihua Xu[2], Hao Huang[1,*], Eng Siong Chng[2]

[1]School of Information Science and Engineering, Xinjiang University, Urumqi, China
[2]School of Computer Science and Engineering, Nanyang Technological University, Singapore

Abstract

In this paper, we propose a single multi-task learning framework to perform End-to-End (E2E) speech recognition (ASR) and accent recognition (AR) simultaneously. The proposed framework is not only more compact but can also yield comparable or even better results than standalone systems. Specifically, we found that the overall performance is predominantly determined by the ASR task, and the E2E-based ASR pretraining is essential to achieve improved performance, particularly for the AR task. Additionally, we conduct several analyses of the proposed method. First, though the objective loss for the AR task is much smaller compared with its counterpart of ASR task, a smaller weighting factor with the AR task in the joint objective function is necessary to yield better results for each task. Second, we found that sharing only a few layers of the encoder yields better AR results than sharing the overall encoder. Experimentally, the proposed method produces WER results close to the best standalone E2E ASR ones, while it achieves 7.7% and 4.2% relative improvement over standalone and single-task-based joint recognition methods on test set for accent recognition respectively.

Index Terms: End-to-end, speech recognition, accent recognition, joint training, multi-task learning

1. Introduction

The advent of End-to-end (E2E) modeling approach in machine learning has fundamentally changed speech community in recent years. The main success of the E2E method lies in its simplicity, compactness, as well as effectiveness. Take Automatic Speech Recognition (ASR) for example, we boil down conventional acoustic, language, as well as lexicon modeling work into an entire network that typically contains two components, that is, the encoder and decoder. Both encoder and decoder are concatenated with attention sub-network and the whole network is optimized with the same objective functions.

Although very successfully applied to wide range of speech processing areas, the potential of E2E modeling framework has yet to be fully unleashed. For instance, one can employ E2E framework to perform diversified speech tasks, such as speaker recognition [1, 2], accent recognition [3], as well as speech recognition [4, 5, 6], just to name a few, aimed at a single task (mono-task) at a time. Intuitively, this is different to what humans are doing. To recognize an incoming speech signal, human can easily perform speech recognition, as well as other recognition, such as speaker identity, gender, and accent recognition etc. simultaneously. In other words, humans are multi-task recognition performer, while present E2E framework are mainly targeted at mono-task recognition, and its multi-task capability is not fully demonstrated.

In this paper, we attempt an E2E-based multi-task learning approach to conduct joint speech and accent recognition simultaneously. Our work is motivated by a workshop participation [7]. There are two tracks of the challenge, that is, 8 accented English speech recognition and corresponding accent classification respectively. Under the E2E modeling framework, both tasks can be easily merged into a single architecture to perform [8]. We are curious if we can still achieve state-of-the-art results on each task with a single multi-task modeling framework. It is not only simpler but is also close to what our humans are doing, namely, performing speech and accent recognition simultaneously.

The main contributions of the work lie in the following aspects. 1) By multi-task learning, we mean our focus is to use a single network architecture to undertake two tasks simultaneously. This differentiates our work from the prior works, since majority of them only focus on accented ASR with multi-task learning [9, 10, 11, 12, 13, 14]. 2) We found ASR pretraining (with or without out-domain data) is essential to achieve better results on either task, particularly for the accent recognition task [15]. 3) By scaling down the accent recognition loss in our combined objective function during training, we can obtain better results on either task consistently.

The rest of this paper is organized as follows. Section 2 describes related works. Section 3 describes the model architecture of the proposed multi-task method and corresponding optimization method. Section 4 describes the overall data for experiments, and Section 5 briefs the experimental setups. Section 6 reports the experimental results, and it is followed by the analysis in Section 7. Finally, we conclude in Section 8.

2. Related work

Accented ASR [16, 17] and accent recognition [18, 19] have been extensively studied recently, but majority of the research works on the two regards are either performed separately or using one to boost another with multi-task method [12, 13, 14, 20]. Assuming an accented English is a product of foreign English influenced by a specific native language, [13] proposed a multi-task learning approach to train an improved accented speech recognition model. Similarly with the multi-task framework, [12] employed accented phoneme-based multi-task learning to boost accented ASR performance, while both [14] and [20] proposed to use accent embedding for improved accented speech recognition. Aside from multi-task learning approach, [9] proposed a learning-based mixture of transform dealing with phone and accent variability in a joint framework. For accent recognition, the work in [3] has employed phone posteriorgram (PPG) features as input to train the accent classifier for the accent challenge track. With diversified data augmentation, as well as system fusion methods, it has achieved the best results out of scores

of participants.

From the above-mentioned, one of the most closely related work to us is [8], which not only employed multi-task learning approach to speech and accent recognition, but also reported the performance for each task, though their main focus is for improved accented speech recognition. The main distinctions are two aspects. First, the contribution of ASR pretraining are not studied in [8, 16, 17], particularly for the efficacy on accent recognition. Second, the objective losses from the two tasks are not analyzed. We found though the two losses are not on the same range, and ASR loss is much bigger. Therefore, scaling down the accent objective loss consistently yields better results for each recognition task.

3. Model architecture and optimization

3.1. Architecture

In this paper, all of our work is performed with Transformer-based E2E framework [21], from the baseline standalone ASR, accent recognition, to the proposed Multi-Task-based Joint Recognition (MTJR). Figure 1 illustrates the entire network architecture for the proposed method.
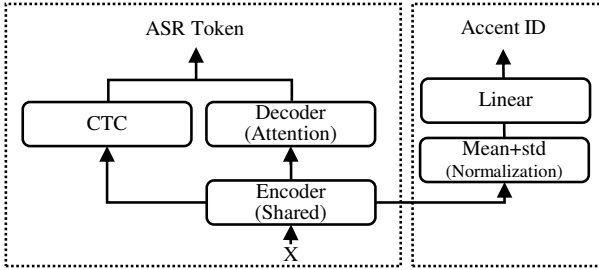

Figure 1: *Multi-task network architecture for joint End-to-End speech recognition and accent recognition.*

From Figure 1, our ASR network is a standard Transformer-based E2E network [22], including encoder, decoder components boosted with self-attention, as well as cross-attention mechanism. We note that the output of the ASR are independent of accent. For the accent recognition network (on the right-hand side of Figure 1), it has a shared encoder with the ASR network. With the outputs from the shared encoder, a pooling layer is employed to estimate the mean and variance vectors. We then concatenate the mean and variance vectors as input to a linear transform followed by a softmax layer to estimate each accent probability.

3.2. Optimization

From Figure 1, the Encoder is jointly optimized with two objective losses, one is from ASR, denoted as $\mathcal{L}_{asr\_loss}$, while the other is from accent recognition, denoted as $\mathcal{L}_{accent\_loss}$; that is, the total objective loss is as Equation 1.

$$\mathcal{L}_{loss} = \beta \mathcal{L}_{asr\_loss} + \lambda \mathcal{L}_{accent\_loss} \qquad (1)$$

where both β and λ are hyper-parameters, and normally they are subject to $\beta + \lambda = 1$. To simplify the tuning process, we train the multi-task network with Equation 2 in practice.

$$\mathcal{L}_{loss} = \mathcal{L}_{asr\_loss} + \lambda \mathcal{L}_{accent\_loss} \qquad (2)$$

where we fix the ASR loss, while keep tuning the loss from accent recognition. This gives us more flexibility to investigate

how the overall performance is impacted by the accent recognition task. We will analyze this in Section 7.2 using Equation 2 to optimize. Meanwhile, the $\mathcal{L}_{asr\_loss}$ in Equation 2 is realized as Equation 3.

$$\mathcal{L}_{asr\_loss} = \gamma \mathcal{L}_{ctc} + (1 - \gamma) \mathcal{L}_{att} \qquad (3)$$

That is, our ASR network is jointly optimized with connectionist temporal classification (CTC) [23] and attention network objective functions, and γ is also a hyper-parameter. In all our following experiments, we found that $\gamma = 0.3$ can get better results.

4. Data description

For the accented ASR and accent recognition challenges in the workshop [7], the organizer allows each participant to use two categories of data, i.e., in-domain and out-domain data. The in-domain data is sponsored by `DataTang` Company in China[1]. It contains 8 accents of English read speech data, that are, American(US), British(UK), Chinese(CHN), Indian(IND), Japanese(JPN), Korean(KR), Portuguese(PT) and Russian(RU) accent respectively. Each accent has roughly 20 hours. The out-domain data is Librispeech with 960 hours, which is optional for participants to use. Besides, there is no accent information available for Librispeech data. Table 1 describes the overall data distributions.

Table 1: *Data description for both accented ASR and accent recognition challenges. The so-called "in-domain" part refers to the required data sets for each participant, while "out-domain" data is optional to use.*

Category	Name	Len (Hrs)	Word/Utt.	Sec./Utt.
In-domain	Train	148.5	9.72	4.29
	Dev	14.5	9.66	4.35
	Test	20.95	9	4.15
Out-domain	Libri.	961	33.43	12.3

5. Experimental setup

All experiments are performed using E2E-based Transformer modeling framework with Espnet tookit [24]. For comparison, we train two groups of models, one is mono-task models that are either for speech recognition or for accent recognition only, and the other is joint models that are realizing two kinds of approaches to conduct joint speech and accent recognition simultaneously. Specifically, one follows the work in [25, 26], which is actually a normal Transformer ASR system but appending each decoded utterance with a recognized accent label, we call it single-task-based joint recognition (STJR) method, and the other is our proposed MTJR method. For speech and accent recognition tasks, the encoders of the Transformer method are the same, while the decoders are different, and it is much simpler in the case of accent recognition task as mentioned in Section 3.1.

We have conducted two kinds of experiments to train the models mentioned above, one is to train them from scratch, and the other is to employ pretraining. For pretraining, we use either in-domain data only, or the overall data that includes both in-domain and out-domain data, to train a Transformer-based ASR

---

[1] https://www.datatang.com/INTERSPEECH2020

Table 2: *Accent recognition accuracy (%) with different Transformer configurations, with or without ASR pretraining using in-domain or out-domain training data sets over `Test` data set in Table 1.*

	System	Avg	US	UK	CHN	IND	JPN	KR	PT	RU
Transformer-3L	S1	30.7	21.1	56.9	31.3	41.6	31.2	9.7	41.1	18.1
Transformer-6L	S2	29.7	15.9	44.2	32.4	33.6	21.4	24.6	35	36
Transformer-12L	S3	32.6	20.8	40.6	22.7	46.2	38.8	36.7	35.2	23.8
ASR-pretrain + S1	S4	45	49.7	73.5	20.1	54.2	45	41.3	43.4	36.1
ASR-pretrain + S2	S5	64.8	55.6	81.4	57.9	82	53.6	64.1	63.3	65.5
ASR-pretrain + S3	S6	66.6	55.4	83.9	67.5	83.9	49.3	76.7	65	55.4
S6 + Libri.	S7	69.8	52	92	73	86.4	55.1	78	70.3	57.8

system. Then we fine-tune the initialized components with in-domain data for the two tasks respectively.

For the actual training, we use the acoustic feature that is 83-dimensional with 80-d filter bank plus 3-d pitch features [27]. The Transformer ASR and the two joint methods are configured with 12-layer encoder, 6-layer decoder with 4-head attention, while the mono-task-based accent recognition method uses three different numbers of layers (3/6/12) in encoder for experimentation. During training, we proceed with 0.1 drop-out, and the Transformers are optimized with Noam optimizer [28]. The output of ASR has 2000 BPEs [29]. Besides, to yield robust models, we adopt various data augmentation methods including spec-augmentation [30], as well as speed perturbation [31].

6. Results

Table 2 reports our mono-task-based accent recognition results using Transformer methods with or without ASR pretraining over `Test` data set in Table 1. From Table 2, we observe bigger Encoder (12-layer) model produces better results. Besides, we obtain significantly better classification results while employing ASR pretraining method. Moreover, it benefits further when using out-domain data (Librispeech) to pretrain the model. Comparing the results between systems with or without pretraining, we observer the best average accuracy (69.8%) of the pretrained system (S7) is more than double of that (32.6%) of the best system (S3) without pretraining at all. Notably, we also get similar results on `Dev` data set, however we skip it due to space limitation.

Table 3 reports the results of both joint training and multi-task learning with (M2, M4, M5, M7, M8) or without (M1, M3, M6) ASR pretraining. For the two joint learning methods, we employ both in-domain-only and the overall (Librispeech included) data based pretraining respectively. To perform the recipe, we first use the data to train a normal Transformer ASR system as indicated in the left-hand side of Figure 1. After that we use the in-domain data to fine-tune the system.

For the systems (M1, M3, and M6) without pretraining recipe in Table 3, we can see the proposed MTJR method achieves both better ASR and better accent recognition results on both `Dev` and `Test` data sets, compared to the STJR method proposed in [25, 26], that is, M3 versus M6, in Table 3. However, the ASR results from both joint recognition methods are worse than those of the mono-task ASR, namely, M3 and M6 versus M1 respectively in Table 3. This suggests the accent recognition task has some negative effects on the ASR performance.

For the systems (M2, M4, M5, M7 and M8) with pretraining method in Table 3, we achieve significant ASR performance improvement in every case. What we observe here is agreed

with the work in [15] which also observed ASR pretraining is essential to improve the results for either mono-lingual, multi-lingual or cross-lingual task. For the ASR results of the 3 methods in Table 3, the proposed MTJR method results slightly worse than those of the Transformer and STJR method respectively. This again reminds us looking into how accent recognition task affects the ASR performance. To put it simply, we conjecture there would be an λ in Eq. 2 or other factors leading to optimal results on either tasks for the MTJR case, which are to be analyzed in Section 7 specifically.

Regarding to the accent recognition results in Table 3, we observe that the proposed MTJR method is not only better than the STJR method without pretraining (M3 versus M6), it also benefits significantly more with pretraining method. It consistently outperforms the STJR method when we compare M4 versus M7, and M5 versus M8 respectively. Particularly, MTJR method achieves 75.2% average accent recognition accuracy while the STJR method only achieves 72.2% accuracy on the `Test` set, when the overall data is used for the pretraining. Such an accent recognition result places us in the second position out of more than 70 participants [7], and the best result [3] is obtained with 60 times of data augmentation as well as feature-based system fusion. we note that we conducted 2 kinds of joint training recipes, that is, either output a accent label before actual ASR output or append an accent label after ASR final output. We have not observed much WER difference from the two setups.

Table 3: *Joint speech and accent recognition results with or without ASR pretraining. All systems use speed perturb and spectral augmentation(specAug) by default. For MTJR, we set $\lambda = 0.1$. M2,M5,M8 use all data to train background model, but M4,M7 use in-domain data only.*

Method	System	WER(%)		ACC(%)	
		Dev	Test	Dev	Test
Transformer	M1	6.9	8	–	–
	M2	5.7	6.7	–	–
STJR [25, 26]	M3	9.1	10.6	80.5	69.7
	M4	7	8	75.7	68.4
	M5	5.8	6.6	77	72.2
MTJR	M6	8.1	9.5	80.2	71
	M7	7.3	8.4	81.8	72.4
	M8	6.2	7.1	82.4	75.2

7. Analysis

As observed from Table 3, though the proposed MTJR method can achieve much better accent recognition results, it's ASR WERs are little bit worse compared to those of the Transformer

or STJR methods. Here, we are curious about in what situation we can achieve desired results for either task.

7.1. Objective losses

To answer the question, it seems that we just simply tune the λ in Eq. 2. However, we want to examine the objective losses versus training epochs for the two tasks first. This gives us more insight about the property of the two tasks. Figure 2 plots the loss versus training epoch for both speech and accent recognition respectively.

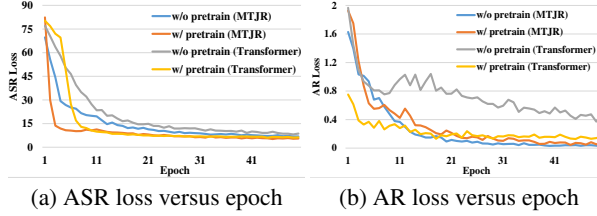(a) ASR loss versus epoch   (b) AR loss versus epoch

Figure 2: *Speech and Accent Recognition losses versus training epoch for MTJR method.*

One of the main points worth our attention from Figure 2 is the ASR loss value is obviously larger than the accent recognition loss value. The latter is so small so as to be negligible compared with the former. Though with small loss value, the accent recognition task still has negative effect on the ASR performance as shown in Table 3. Next, we see how to obtain optimal results by means of λ tuning.

7.2. Tuning λ for optimal results

Figures 3 and 4 plot speech and accent recognition results versus λ in Eq. 2 on `Test` set respectively for the MTJR method.
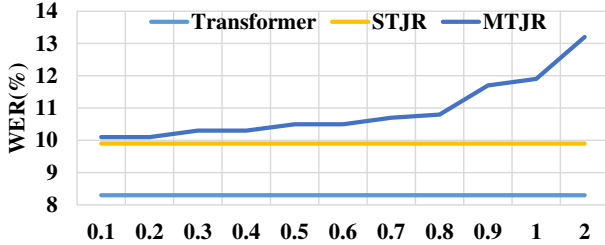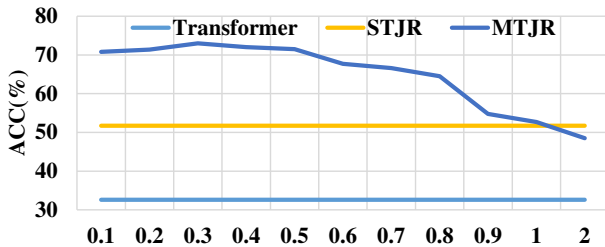

Figure 3: *Test set WER versus λ for MTJR method.*

Figure 4: *Test set accent recognition accuracy versus λ for MTJR method.*

From either Figure, we see that as λ increases, either task yields degraded results though the accent loss is rather small as

shown in Figure 2. Specifically, we found that keeping λ around [0.1, 0.3] can let us have better results on both tasks.

7.3. Multi-task encoder-layer sharing

It is a common interpretation that the initial layers of the encoder are expected to encode lower level acoustic information, while subsequent layers codify more complex cues closer to phonetic classes [32]. For E2E-based speech recognition, one can reasonably think encoder is responsible for feature learning while decoder is for linguistic interpretation. However, prior work [33] even shows different layers of encoder are learning for different tasks. As a result, it naturally comes up with a question: which layer sharing can yield better results for speech or accent recognition. Table 4 reports the experimental results with different encoder layers being shared between speech recognition and accent recognition tasks.

Table 4: *The results of MTJR method by different encoder layers sharing with $\lambda = 0.1$ in Eq. 2.*

Shared layers	System	WER(%)		ACC(%)	
		Dev	Test	Dev	Test
Layer-3	L1	8.7	10	64.8	54.6
Layer-6	L2	8.6	10.1	78.3	66
Layer-9	L3	8.7	9.9	80.2	71.4
Layer-12	L4	8.6	10.1	77.1	70.8

We see from Table 4 the best recognition results are obtained when 9 layers of the overall 12-layer encoder are shared. With a fixed λ, the WER results are not obviously affected by how many layers are shared. However, the accent recognition results are different. Only lower layer sharing (L1 and L2 in Table 4) yields much worse results, while overall layer sharing (L4) does not mean the best results. We note that results in Table 4 cannot be compared with what are presented in Table 3, since speed perturbation is employed in Table 3 while it is not in Table 4 for simplicity.

8. Conclusions

In this paper we proposed an E2E-based multi-task learning approach to joint speech and accent recognition. The benefit of the method is it can perform speech and accent recognition simultaneously. We verify its effectiveness on an accented English data set that is released for speech and accent recognition contests in a workshop. Compared to a conventional mono-task Transformer ASR system, the ASR performance is slightly degraded, while its performance on the accent recognition is noteworthy, especially when the mutli-task system is initialized with ASR pretraining using in-domain and out-domain data. Besides, we conducted a series of analysis work. We found though the objective loss of the accent recognition task is much smaller, scaling down it in the joint objective function yields better results for both recognition tasks. Finally, we also investigated different encoder-layer sharing for the multi-task method, and found partial sharing (at 9th layer), instead of the overall layer (12 layers) of the encoder, produced the best accent recognition results.

9. Acknowledgements

10. References

[1] Soroosh Tayebi Arasteh, "Generalized LSTM-based end-to-end text-independent speaker verification," *arXiv preprint arXiv:2011.04896*, 2020.

[2] Suwon Shon, Hao Tang, and James Glass, "Frame-level speaker embeddings for text-independent speaker recognition and analysis of end-to-end model," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 1007–1013.

[3] Houjun Huang, Xu Xiang, Yexin Yang, Rao Ma, and Yanmin Qian, "AISPEECH-SJTU accent identification system for the Accented English Speech Recognition Challenge," *arXiv preprint arXiv:2102.09828*, 2021.

[4] Yotaro Kubo and Michiel Bacchiani, "Joint phoneme-grapheme model for end-to-end speech recognition," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6119–6123.

[5] Binbin Zhang, Di Wu, Zhuoyuan Yao, Xiong Wang, Fan Yu, Chao Yang, Liyong Guo, Yaguang Hu, Lei Xie, and Xin Lei, "Unified streaming and non-streaming two-pass end-to-end model for speech recognition," *arXiv preprint arXiv:2012.05481*, 2020.

[6] Genta Indra Winata, Samuel Cahyawijaya, Zihan Liu, Zhaojiang Lin, Andrea Madotto, Peng Xu, and Pascale Fung, "Learning fast adaptation on cross-accented speech recognition," *arXiv preprint arXiv:2003.01901*, 2020.

[7] Xian Shi, Fan Yu, Yizhou Lu, Yuhao Liang, Qiangze Feng, Daliang Wang, Yanmin Qian, and Lei Xie, "The accented English speech recognition challenge 2020: open datasets, tracks, baselines, results and methods," *arXiv preprint arXiv:2102.10233*, 2021.

[8] Thibault Viglino, Petr Motlicek, and Milos Cernak, "End-to-end accented speech recognition.," in *INTERSPEECH*, 2019, pp. 2140–2144.

[9] Abhinav Jain, Vishwanath P Singh, and Shakti P Rath, "A multi-accent acoustic model using mixture of experts for speech recognition,," in *INTERSPEECH*, 2019, pp. 779–783.

[10] Han Zhu, Li Wang, Pengyuan Zhang, and Yonghong Yan, "Multi-accent adaptation based on gate mechanism," *arXiv preprint arXiv:2011.02774*, 2020.

[11] Xuesong Yang, Kartik Audhkhasi, Andrew Rosenberg, Samuel Thomas, Bhuvana Ramabhadran, and Mark Hasegawa-Johnson, "Joint modeling of accents and acoustics for multi-accent speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1–5.

[12] Kanishka Rao and Haşim Sak, "Multi-accent speech recognition with hierarchical grapheme based models," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 4815–4819.

[13] Shahram Ghorbani and John HL Hansen, "Leveraging native language information for improved accented speech recognition," *arXiv preprint arXiv:1904.09038*, 2019.

[14] Abhinav Jain, Minali Upreti, and Preethi Jyothi, "Improved accented speech recognition using accent embeddings and multi-task learning," in *Interspeech*, 2018, pp. 2454–2458.

[15] Jocelyn Huang, Oleksii Kuchaiev, Patrick O'Neill, Vitaly Lavrukhin, Jason Li, Adriana Flores, Georg Kucsko, and Boris Ginsburg, "Cross-language transfer learning, continuous learning, and domain adaptation for end-to-end automatic speech recognition," *arXiv preprint arXiv:2005.04290*, 2020.

[16] Shahram Ghorbani, Ahmet E Bulut, and John HL Hansen, "Advancing multi-accented LSTM-CTC speech recognition using a domain specific student-teacher learning paradigm," in *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2018, pp. 29–35.

[17] Han Zhu, Li Wang, Pengyuan Zhang, and Yonghong Yan, "Multi-accent adaptation based on gate mechanism," in *Interspeech 2019*, 2019.

[18] Kay Berkling, Marc A Zissman, Julie Vonwiller, and Christopher Cleirigh, "Improving accent identification through knowledge of English syllable structure," in *Fifth International Conference on Spoken Language Processing*, 1998.

[19] Yishan Jiao, Ming Tu, Visar Berisha, and Julie 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.

[20] Bo Li, Tara N Sainath, Khe Chai Sim, Michiel Bacchiani, Eugene Weinstein, Patrick Nguyen, Zhifeng Chen, Yanghui Wu, and Kanishka 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.

[21] Linhao Dong, Shuang Xu, and Bo 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.

[22] Suyoun Kim, Takaaki Hori, and Shinji Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 4835–4839.

[23] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen 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.

[24] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, et al., "Espnet: End-to-end speech processing toolkit," *arXiv preprint arXiv:1804.00015*, 2018.

[25] Wenxin Hou, Yue Dong, Bairong Zhuang, Longfei Yang, Jiatong Shi, and Takahiro Shinozaki, "Large-scale end-to-end multilingual speech recognition and language identification with multi-task learning," *Babel*, vol. 37, no. 4k, pp. 10k, 2020.

[26] Shinji Watanabe, Takaaki Hori, and John 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)*. IEEE, 2017, pp. 265–271.

[27] Pegah Ghahremani, Bagher BabaAli, Daniel Povey, Korbinian Riedhammer, Jan Trmal, and Sanjeev Khudanpur, "A pitch extraction algorithm tuned for automatic speech recognition," in *2014 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2014, pp. 2494–2498.

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[29] Rico Sennrich, Barry Haddow, and Alexandra Birch, "Neural machine translation of rare words with subword units," *arXiv preprint arXiv:1508.07909*, 2015.

[30] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *arXiv preprint arXiv:1904.08779*, 2019.

[31] Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, "Audio augmentation for speech recognition," in *Proc. of INTERSPEECH 2015*, 2015.

[32] Yoshua Bengio, Aaron Courville, and Pascal Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[33] Zhiyun Fan, Meng Li, Shiyu Zhou, and Bo Xu, "Exploring wav2vec 2.0 on speaker verification and language identification," *arXiv preprint arXiv:2012.06185*, 2020.