Adaptive Attention Network with Domain Adversarial Training for Multi-Accent Speech Recognition

Yanbing Yang[1], *Hao Shi*[2], *Yuqin Lin* [1], *Meng Ge*[1], *Longbiao Wang*[1,3], *Qingzhi Hou*[1], *Jianwu Dang*[1,4]

[1] Tianjin Key Laboratory of Cognitive Computing and Application,
College of Intelligence and Computing, Tianjin University, Tianjin, China
[2]Graduate School of Informatics, Kyoto University, Kyoto, Japan
[3] Huiyan Technology (Tianjin) Co., Ltd, Tianjin, China
[4] Japan Advanced Institute of Science and Technology, Ishikawa, Japan

{yangyanbing, longbiao_wang }@tju.edu.cn

Abstract

Spoken accents severely degrade the performance of automatic speech recognition (ASR) systems. Domain adversarial training (DAT) is widely adopted for generating domain-invariant features to reduce the influence of accents. However, the generated features trained by DAT are still maintaining some accent discrimination information, limiting the ASR performance. In addition, the features generated by DAT of each accent have different degrees of residual accent discriminant information. In this paper, we propose an adaptive attention network with DAT to further eliminate the influence of retaining accent information in features generated by DAT. We employ the adaptive attention module to transform the encoder output to a more general representation. Experiments on the AESRC2020 dataset show that the proposed method can achieve satisfactory performance improvements on seen and unseen accent when the correct accent information is still preserved in the output of the encoder.

Index Terms: speech recognition, accented speech recognition, domain adversarial training

1. Introduction

Automatic speech recognition (ASR) aims to get transcription from speech. It has achieved remarkable performance in many broadcasting recording scenarios. However, the speech variability problem in the real world poses a serious challenge to the ASR systems. The accent is a typical issue of speech variability [1], which is caused by geographical region, social group, and so on. The performance gap between different accented ASR remains large, thus ASR systems trained on one accent or standard speech degrades when faced with other accented speech. Hence, it is hard to build an accent-robust system with limited accented speech data.

Previous work explores building accent-robust ASR systems in mainly two ways: introducing accent-dependent information and generating accent-independent features. The main idea of introducing accent-dependent information is to use accent-dependent information, such as i-vectors [2], accent IDs [3], or accent embeddings [4, 5], to manage multi-accent ASR systems, or used it in the multi-task [6, 7] manner. Some accent adaptive networks also introduce accent-related information [8–10]. Those architectures aim to incorporate accent information into a single generic model, they always achieve satisfactory results in seen accents, while the accent-dependent adaptive networks also can reduce the mismatch between the training data and the test data.

For generating accent-independent features, the ASR systems often make the output of the acoustic model contain as little accent information as possible. Adversarial training is effective to mitigate the accent mismatch problem. Domain adversarial training (DAT) [11] is a common technique of adversarial training, which enforces intermediate representations to be domain-invariant for different accented speech. It has been shown to improve the accent robustness of multi-accent ASR models [12–15]. DAT attempts to remove the accent information from the output of the end-to-end (E2E) ASR encoder (often playing the role of an acoustic model). However, we find that the output of the encoder trained with DAT still has some residual accent discrimination information, which makes the DAT ASR model performance bad on some accents.

In this paper, to eliminate the influence of residual accents information, we propose the Adaptive Attention Network with Domain Adversarial Training (AANet) method. AANet adopts DAT to initial the transformer encoder. Experimental results show AANet can boost speech recognition in many accents, and especially can improve the performance in unseen accents, further reducing the accent mismatch in the DAT-trained model. In AANet, the adaptive attention module transforms the output of the encoder into adaptive features and inputs them to the transformer decoder, and the attention-based adaptor acquires accent information through an accent classifier.

2. DAT for Multi-Accent ASR

2.1. Transformer-based E2E ASR

The transformer is an E2E architecture [16], consisting of the multi-layer encoder, and multi-layer decoder. The encoder and decoder layer are boosted with self-attention, as well as a cross-attention mechanism. The encoder plays a role as an acoustic model, and the output of the encoder is input to the CTC layer or decoder to get CTC or attention-based results. The CTC and attention are trained simultaneously with CTC-attention joint loss [17], the whole loss function L_{ASR} is as follow:

$$L_{ASR} = (1-\gamma)L_{ATT} + \gamma L_{CTC} \quad (1)$$

where L_{CTC} and L_{ATT} are the CTC and attention network objective losses, respectively. The γ is a hyper-parameter that denotes the weight of CTC loss.

2.2. Domain Adversarial Training

Domain adversarial training (DAT) [11] has been widely applied to ASR systems under multiple conditions like speakers

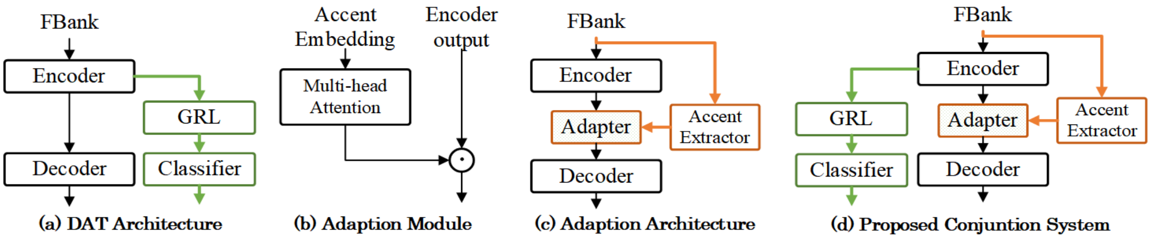

Figure 1: *Flowchart of (a) DAT architecture; (b) proposed adaptive attention module; (c) adaptive attention with E2E baseline; (d) proposed AANet.*

[18–21], noises [22,23], accents [12–14], and languages [24]. It aims to learn an intermediate latent feature space that is domain-invariant. Our domain adversarial framework for speech recognition is illustrated in Fig.1(a), and consists of three main components: the attention encoder $G(x, \theta_f)$, with input speech feature x and parameters θ_f. Accent classification network $C(f_a, \theta_c)$ with input feature f_a and parameters θ_c, f_a is generated by inputting the encoder output feature f to the $mean{+}std$ pooling layer, $f_a = f_{mean} + f_{std}$, and the there is a gradient inversion layer between encoder and classifier. The attention decoder $D(f, \theta_y)$ with input f and parameters θ_y, where y is the inferred transcription of ASR model. The DAT objective function is written as follows:

$$E(\theta_f, \theta_y, \theta_c) = L_{ASR}(\theta_f, \theta_y) + L_C(\theta_f, \theta_c) \qquad (2)$$

where L_{ASR} denotes the ASR prediction loss function, and L_C denotes a cross-entropy loss function for the accent classification network. Denote the weight matrices of G, C, D as $\theta_f, \theta_c, \theta_y$. And the network optimization strategy is as follows:

$$\begin{aligned}(\hat{\theta}_f, \hat{\theta}_y) &= \arg\min_{\theta_f, \theta_y} E(\theta_f, \theta_y, \hat{\theta}_c) \\ \hat{\theta}_c &= \arg\max_{\theta_c} E(\hat{\theta}_f, \hat{\theta}_y, \theta_c)\end{aligned} \qquad (3)$$

The “min-max” optimization of DAT is done simultaneously within a single backward pass by using by applying the gradient reversal layer between the generator G and accent classifier C. Each weight is updated by the following gradient descent rules:

$$\theta_f \leftarrow \theta_y - \alpha\left(\frac{\partial L_{ASR}}{\partial \theta_f} - \lambda \frac{\partial L_C}{\partial \theta_f}\right) \qquad (4)$$

$$\theta_c \leftarrow \theta_c - \alpha \frac{\partial L_C}{\partial \theta_f} \qquad (5)$$

$$\theta_y \leftarrow \theta_y - \alpha \frac{\partial L_{ASR}}{\partial \theta_f} \qquad (6)$$

Where α is the learning rate and λ is the scale of L_C gradients, adjusting $\lambda > 0$ to experiment with DAT.

3. Adaptive Attention Network with DAT

DAT can not completely eliminate the accent discriminant information, which restrains the ASR performance, we demonstrate this by experimental results in section 5. To mitigate the drawbacks, we proposed a novel method, Adaptive Attention Network with DAT (AANet), to further remove the mismatches between accents through the adaptive attention module. The architecture of AANet is based on the E2E DAT architecture, with an accent embedding extraction network and the novel adaptive attention module. In this section, we describe each module in detail and the corresponding training strategies of AANet.

3.1. Accent Embedding Extraction Network

The accent extractor network is built according to [25], which is a multi-layer encoder based on self-attention (SA). We use the output of the last encoder as accent embedding v_a ($a \in U$, U is the set of accents). For accent classify, a mean + std pooling layer is applied after the last encoder to pool the output on T dimension, after pooling, we input it into a linear layer to classify the accent and optimize the CE loss:

$$L_E = CE(Linear(mean(v_a) + std(v_a)), v_{true}) \qquad (7)$$

where v_{true} is the ground truth. Besides, we also use the encoder trained by the ASR downstream task to initialize the encoder of the accent classification network.

3.2. Adaptive Attention Module

The architecture of the adaptive attention module is shown in Fig. 1(b) and 1(c) depicts the structure of the adaptive attention network when it is used in a simple E2E network. The adaptive module is referred to as $\mathcal{A}_{att}$. The MHA represents the multi-head self-attention network, which allows the network to jointly attend to accent information from different representations subspaces. Fig. 1(b) depicts the structure of the adaptive attention network when it is used in a simple E2E network. The entire adaptation process is described as the following formula:

$$\begin{aligned}\mathcal{A}_{att}(f, v_a) &= f \odot SigMoid(MHA(Q, K, V)) \\ Q = v_a, W^Q&, K = v_a W^K, V = v_a W^V\end{aligned} \qquad (8)$$

where $\odot$ denotes the element-wise product, W^K, W^Q and W^V are the weight matrixs that transform v_a into Q, K, V.

3.3. Training Strategy of AANet

We apply the adaptive attention module proposed in section 3.2 to optimize the speech recognition performance trained by DAT one step further.

The novel architecture of the proposed model is shown in Fig. 1(d). The training strategy is as follows:
(1) We pretrain the accent extractor according to 3.1. It is used to output stable accent discrimination embeddings.
(2) We freeze the parameters of the accent extractor and adapter. Then we train the E2E-ASR network by DAT until the E2E network converges.
(3) We freeze the parameters of the accent discriminator and the accent extractor. Then train the DAT pre-trained encoder and decoder with the attention-based adapter.

4. Experimental Settings

4.1. Dataset

We conducted experiments on the dataset of Accented English Speech Recognition Challenge 2020 (AESRC2020) [25], which contains a training set for 8 English accents in England (UK), America (US), China (CHN), Japan (JPN), Russia (RU), India (IND), Portugal (PT), and Korea (KR), with 20-hour for each accent. There are two more accents Canada (CAN) and Spain (ES) are included in the test set. We report the word error rate (WER) on the test sets. The training, development, and test set contain 148.5 hours, 14.5 hours, and 20.95 hours of data respectively.

4.2. Features and Network Settings

In all experiments, we use the 80-dimensional Mel-filterbank feature as the input of the acoustic model and the frame length is 25 ms with a 10 ms shift. the 1000 English Byte Pair Encoding (BPE) subword units are adopted. For the RNN language model, the 1000 English Byte Pair Encoding (BPE) [26] subword units are adopted. For the E2E ASR baseline, we adopt the transformer with the configuration of a 12-layer encoder and a 6-layer decoder, where each self-attention layer has an attention dimension of 256 and 4 heads, following the settings of the official baseline [25]. SpecAugment [27] is also applied for data augmentation. During decoding, the CTC module is used for score interpolation [17] with a weight of 0.3, and a beam-width of 10 is applied for beam searching. All models are built using the ESPnet toolkit [28]. Table 1 presents the performance of our baseline and the officially provided baseline on the dev set.

Table 1: *Recognition performance (WERs) (%) of our baseline and the offical provided baseline on dev set.*

Baseline	**CHN**	**IND**	**RU**	**JPN**	**PT**	**UK**	**KR**	**US**	**AVE**
Official	11.77	10.05	5.26	6.79	7.45	10.06	7.69	9.96	8.63
Ours	12.37	9.09	4.76	6.83	7.60	9.89	7.95	9.51	8.55

In the DAT architecture, we chose the last block of the encoder to generate the domain-invariant feature, the generator output will be sent to the mean + std pooling and then the 256×8 linear layer to classify accent. We experiment with DAT by keeping $\lambda = 0.004$, which makes the DAT network achieve the best performance.

Table 2: *Accent accuracy of the accent embedding extractor classification on 8 accents in the test set.*

	US	**UK**	**KR**	**PT**	**JPN**	**RU**	**IND**	**CHN**
Acc(%)	71.86	72.25	47.01	42.17	56.02	43.12	80.93	48.68

In the architecture of adaptive attention with simple E2E and the AANet, the accent extractor has a 12-layer encoder, each self-attention layer has an attention dimension of 256 and 4 heads, and the adaptive attention module uses a 4-heads self-attention. Table 2 shows the accent extractor classification accuracy on 8-seen accents of the test set. The DAT settings are the same as the DAT architecture.

5. Experimental Results

5.1. Analysis of DAT-based Feature

We observe the performance of accent classification according to three training steps of DAT. Table 3 shows the accent accuracy in different stages of DAT. The $S1$, $S2$, and $S3$ denote the accent classifier optimization stage, ASR and accent co-optimization stage, and accent-invariant feature generating stage respectively. In $S1$, the network mainly optimizes the accent classification task. The accent classifier obtains 40.5% accuracy. In $S2$, the network learning targets are ASR accuracy and accent classifier accuracy optimization. The accent accuracy has been increased to approximately 60.2%, indicating the presence of sufficient accent discrimination information in the encoder-generated features. In $S3$, the main task of the network is to promote the generator to generate domain-invariant features through adversarial training. At this time, the accent classifies accuracy drops from 60.2% to 38.5%, indicating that the accent information in the generated features is gradually reduced. There are 8 different accents in the training set of the 2020AESRC dataset, meaning the classification accuracy should be around 12.5% when features generated by the encoder become completely accent irrelevant. 38.5% accuracy means the feature still has a small amount of accent information.

Table 3: *Accent accuracy of the accent classification in different stages of DAT, on the test set of 2020AESRC seen accents.*

	$S1$	$S2$	$S3$
Acc(%)	40.5%	60.2%	38.5%

5.2. Effects of DAT

The experimental results of word error rates (WERs) are shown in Table 4. We see that DAT significantly boosts the performance across all accents, even those unseen accents. This shows that the accent-invariant features help to improve the accent robustness.

5.3. Effects of AANet for Seen Accents

In Table 5, we statistic the accent accuracy of the n-th largest softmax value in the DAT model to analyze the effect of DAT on accent removal. We count the frequencies of the different accents in the test set on the n-th largest softmax value, then divide it by the total number of corresponding accents to get the accuracy of the n-th softmax. We got the following conclusion:

(1) The output of the encoder still retains the correct accent information:

US, UK, KR, and PT still had a part of the correct accent information after DAT. We used accent embeddings instead of hard accent categories to obtain accent information. It can be seen from Table 5 that the maximum accuracy after softmax of US and KR was higher than 12.5%, which indicates that they still contain some correct accent information. For UK and PT, even though their softmax maximum prediction was not accurate enough, they have high accuracy on the second-largest softmax value. This provides them with enough correct accent information. From the results in Table 4, it shows that our proposed AANet had a high improvement in ASR performance when handling samples that have partial residual correct accent information. Combined with Table 2, the effect of the accent extractor also affects the adaptive results. Besides, Fig. 2 is the confusion matrix of the maximum softmax value after DAT. All accents (except IND) were incorrectly classified as IND. US, UK, KR, and PT achieved the second-highest accuracy rate in each corresponding correct accent category. For the correct partial accent information contained in the network, the adaptive method eliminates the influence of other wrong accents.

(2) The output of the encoder contains incorrect accent information:

JPN and RU contained a lot of wrong accent information. It can be seen from Table 5 that their softmax top 4 accuracy rates were very small. Besides, accent classifiers do not provide enough accent information: JPN got 56.02% and RU got

Table 4: *Recognition performance (WERs) (%) for different systems compute across multiple accents in test set. AA denotes the adaptive attention module.*

	Seen Accent								*Unseen Accent*	
Approach	**CHN**	**IND**	**RU**	**JPN**	**PT**	**UK**	**KR**	**US**	**ES**	**CAN**
AANet(DAT+AA)	16.09	**10.43**	**10.56**	**5.59**	**6.70**	**6.06**	**5.98**	**7.37**	**9.60**	**7.23**
DAT	**15.78**	10.47	10.73	5.70	7.08	6.44	6.37	7.87	9.82	7.66
Baseline+AA	16.24	11.39	11.21	6.14	7.73	7.08	6.72	8.44	10.28	8.25
Baseline	16.27	11.96	12.01	6.42	8.16	7.38	6.70	8.90	12.40	8.98

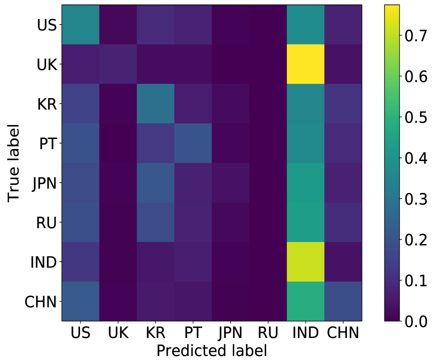

Figure 2: *Confusion matrix of DAT accent classification results.*

43.12% accuracy in Table 2 respectively. Fig. 2 shows that the two accents are both lower than the other ones, and had a similar distribution in the classification confusion matrix, even though they are different accents. This shows that DAT will process some features of different accents into similar features. For JPN, there is already a relatively small WER. But for RU, it is difficult to improve the performance. For the incorrect accent information contained in the network, the adaptive method has a general boost.

(3) The output of the encoder still contains a lot of accent information:

Table 5 shows the high classification accuracy of IND accents, which suggests the IND accent contains a lot of accent distinguishing information. This means that adding accent information to the adaptation has little effect.

(4) The output of the encoder has less accent information retaining:

Compared with other accents in Table 2, CHN is the most accent-invariant accent. Table 5 shows that different softmax values of CHN are very similar and not far from 12.5%. Adaptation based on the fact that the accent information is already stripped off will degrade the ASR performance.

Table 5: *The accent accuracy of the n-th largest softmax value in DAT model (test set). S_n represents n-th largest softmax.*

Accent	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8
US	0.354	0.223	0.166	0.112	0.077	0.052	0.015	0.001
UK	0.074	0.383	0.195	0.123	0.105	0.070	0.050	0.001
KR	0.285	0.236	0.184	0.155	0.087	0.042	0.011	0.000
PT	0.202	0.311	0.222	0.141	0.075	0.037	0.011	0.000
JPN	0.038	0.101	0.153	0.178	0.212	0.200	0.119	0.000
RU	0.000	0.001	0.000	0.002	0.004	0.008	0.018	0.968
IND	0.712	0.124	0.081	0.042	0.023	0.014	0.046	0.000
CHN	0.184	0.191	0.135	0.123	0.100	0.105	0.159	0.000

5.4. Effects of AANet for Unseen Accents

Although the accent classifier does not contain unseen data, it will find a relationship with the visible training data. Table 6 shows the n-th largest softmax value in DAT model and Table 7 shows the accent extractor classification results. CAN and US have a strong similarity with an accent recognizer of 72.91%. Similar to the US effect on the seen dataset, our proposed method can remove residual accent distinguishing information, so its recognition rate is improved. For ES, although there is also high accuracy on US accents, the remaining classes have smoother similarities. In other words, there is not enough information about the correct accent. So the improvement in general.

Table 6: *The unseen accent accuracy of the n-th largest softmax value in DAT model (test set). S_n represents n-th largest softmax.*

	CAN			*ES*		
Accent	S_1	S_2	S_3	S_1	S_2	S_3
US	0.729	0.199	0.048	0.333	0.285	0.173
UK	0.053	0.233	0.205	0.072	0.091	0.122
KR	0.054	0.071	0.085	0.039	0.036	0.060
PT	0.018	0.062	0.095	0.098	0.109	0.123
JPN	0.023	0.066	0.091	0.067	0.062	0.062
RU	0.065	0.179	0.141	0.147	0.144	0.120
IND	0.011	0.071	0.180	0.104	0.150	0.217
CHN	0.048	0.121	0.155	0.140	0.122	0.123

Table 7: *The probability that the accent extractor classifies the unseen accents in the test set as each seen accent*(%).

	US	**UK**	**KR**	**PT**	**JPN**	**RU**	**IND**	**CHN**
CAN	72.91	5.30	5.36	1.80	2.28	6.47	1.06	4.83
ES	33.33	7.22	3.91	9.81	6.72	14.66	10.36	13.99

6. Conclusions and Future Works

In this paper, we explored domain adversarial training (DAT) and proposed an accent adapter to further eliminate the influence of retaining accent information in DAT. We pre-trained the transformer encoder with DAT. Then, the encoder output and accent embedding were input to the adapter to get adaptive features. We found that DAT handles each accent differently: most of the encoder outputs contain residual current accent information; some of the encoder outputs contain incorrect accent information; while a few encoder outputs contain abundant accent information or less accent information retaining. The proposed method can boost ASR performance when the output of the encoder still retains the correct accent information. In the future, we will work on a more fine-grained accent adaption to improve the accent robustness of ASR.

7. ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China (Grant No. 62176182).

8. References

[1] A. Koenecke, A. Nam, E. Lake, J. Nudell, M. Quartey, Z. Mengesha, C. Toups, J. R. Rickford, D. Jurafsky, and S. Goel, "Racial disparities in automated speech recognition," *Proceedings of the National Academy of Sciences*, vol. 117, no. 14, pp. 7684–7689, 2020.

[2] Z. Meng, H. Hu, J. Li, C. Liu, Y. Huang, Y. Gong, and C.-H. Lee, "L-vector: Neural label embedding for domain adaptation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7389–7393.

[3] 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.

[4] M. Najafian, A. DeMarco, S. Cox, and M. Russell, "Unsupervised model selection for recognition of regional accented speech," in *Fifteenth annual conference of the international speech communication association*, 2014.

[5] T. Viglino, P. Motlicek, and M. Cernak, "End-to-end accented speech recognition." in *Interspeech*, 2019, pp. 2140–2144.

[6] Y. Huang, D. Yu, C. Liu, and Y. Gong, "Multi-accent deep neural network acoustic model with accent-specific top layer using the kld-regularized model adaptation," in *Fifteenth Annual Conference of the International Speech Communication Association*. Citeseer, 2014.

[7] A. Jain, M. Upreti, and P. Jyothi, "Improved accented speech recognition using accent embeddings and multi-task learning." in *Interspeech*, 2018, pp. 2454–2458.

[8] K. Rao and H. 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.

[9] A. Das, K. Kumar, and J. Wu, "Multi-dialect speech recognition in english using attention on ensemble of experts," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6244–6248.

[10] H. Zhu, L. Wang, P. Zhang, and Y. Yan, "Multi-accent adaptation based on gate mechanism," *Proc. Interspeech 2019*, pp. 744–748, 2019.

[11] 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.

[12] S. Sun, C.-F. Yeh, M.-Y. Hwang, M. Ostendorf, and L. Xie, "Domain adversarial training for accented speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4854–4858.

[13] H. Hu, X. Yang, Z. Raeesy, J. Guo, G. Keskin, H. Arsikere, A. Rastrow, A. Stolcke, and R. Maas, "Redat: Accent-invariant representation for end-to-end asr by domain adversarial training with relabeling," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6408–6412.

[14] N. Das, S. Bodapati, M. Sunkara, S. Srinivasan, and D. H. Chau, "Best of both worlds: Robust accented speech recognition with adversarial transfer learning," *arXiv preprint arXiv:2103.05834*, 2021.

[15] Y.-C. Chen, Z. Yang, C.-F. Yeh, M. Jain, and M. L. Seltzer, "Aipnet: Generative adversarial pre-training of accent-invariant networks for end-to-end speech recognition," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6979–6983.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[17] S. Kim, T. Hori, and S. 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.

[18] Z. Meng, J. Li, Z. Chen, Y. Zhao, V. Mazalov, Y. Gong, and B.-H. Juang, "Speaker-invariant training via adversarial learning," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5969–5973.

[19] Z. Meng, Y. Zhao, J. Li, and Y. Gong, "Adversarial speaker verification," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6216–6220.

[20] C. Luu, P. Bell, and S. Renals, "Channel adversarial training for speaker verification and diarization," in *2020 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Institute of Electrical and Electronics Engineers (IEEE), 2020, pp. 7094–7098.

[21] Y. Tu, M.-W. Mak, and J.-T. Chien, "Variational domain adversarial learning for speaker verification." in *Interspeech*, 2019, pp. 4315–4319.

[22] A. Beutel, J. Chen, Z. Zhao, and E. H. Chi, "Data decisions and theoretical implications when adversarially learning fair representations," *arXiv preprint arXiv:1707.00075*, 2017.

[23] Y. Shinohara, "Adversarial multi-task learning of deep neural networks for robust speech recognition," *Interspeech 2016*, pp. 2369–2372, 2016.

[24] J. Yi, J. Tao, Z. Wen, and Y. Bai, "Adversarial multilingual training for low-resource speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4899–4903.

[25] X. Shi, F. Yu, Y. Lu, Y. Liang, Q. Feng, D. Wang, Y. Qian, and L. Xie, "The accented english speech recognition challenge 2020: Open datasets, tracks, baselines, results and methods," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6918–6922.

[26] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 66–75.

[27] 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," *Proc. Interspeech 2019*, pp. 2613–2617, 2019.

[28] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N.-E. Y. Soplin, J. Heymann, M. Wiesner, N. Chen *et al.*, "Espnet: End-to-end speech processing toolkit," *Proc. Interspeech 2018*, pp. 2207–2211, 2018.