Improving Distortion Robustness of Self-supervised Speech Processing Tasks with Domain Adaptation

Kuan Po Huang[1⋆] *Yu-Kuan Fu*[2⋆] *Yu Zhang*[3⋆] *Hung-yi Lee*[4⋆]

[1]Graduate Institute of Computer Science and Information Engineering, National Taiwan University
[2]Department of Physics, National Taiwan University
[3]Google Brain
[4]Graduate Institute of Communication Engineering, National Taiwan University
[124]{f09922005, b07202024, hungyilee}@ntu.edu.tw, [3]ngyuzh@google.com

Abstract

Speech distortions are a long-standing problem that degrades the performance of supervisely trained speech processing models. It is high time that we enhance the robustness of speech processing models to obtain good performance when encountering speech distortions while not hurting the original performance on clean speech. In this work, we propose to improve the robustness of speech processing models by domain adversarial training (DAT). We conducted experiments based on the SUPERB framework on five different speech processing tasks. In case we do not always have knowledge of the distortion types for speech data, we analyzed the binary-domain and multi-domain settings, where the former treats all distorted speech as one domain, and the latter views different distortions as different domains. In contrast to supervised training methods, we obtained promising results in target domains where speech data is distorted with different distortions including new unseen distortions introduced during testing.

Index Terms: domain adversarial training, self-supervised models, speech processing tasks, continual training, SUPERB

1. Introduction

Deep learning-based models are powerful in many speech-related applications [1]. However, studies have shown that the domain mismatch problem degrades the performance of acoustic models significantly. Common domain mismatch scenarios include accented speech, multi-lingual speech, or distorted speech, and this paper focuses on distorted speech. In the past, there are some rule-based or statistical-based methods for reducing noise [2–4]. There are also some deep learning based approaches [5, 6] that try to solve this problem. These approaches aim to recover clean speech from noisy speech, but can not guarantee good performance on unseen domain data.

Self-supervised models, which can be trained with large scale unlabeled data, have become a trend with their ability to output better representations for downstream tasks compared to traditional supervised models. Recently, self-supervised models [7–9] trained with target domain speech proved that they could be robust under domain mismatch scenarios. Performing continual training on self-supervised models [10, 11] with unlabeled target domain data also successfully improved performance on downstream tasks.

Another solution to domain mismatch is training models with domain adversarial training [12] (DAT) based on domain adversarial neural networks (DANN). DANN is a combination of a feature extractor, a downstream model, and a jointly trained domain classifier. The domain classifier distinguishes between the source domain and target domain. With this approach, it is believed that the output representations of the feature extractor can be domain-invariant, so the downstream model can perform comparable results in both source and target domains. Some existing works adopt DAT to solve the domain mismatch problem. For example, automatic speech recognition models [13–16] trained with DAT helped dealt with accented speech. DAT also improved performance for multi-lingual speech emotion recognition [17], text-to-speech synthesis [18], speaker recognition [19, 20], and noise adaptive speech enhancement [21].

Previous works have shown that DAT is useful in many cases. However, they lack comprehensive studies across multiple self-supervised speech processing tasks. In our work, we combined the concept of DAT and self-supervised training. We proposed to adopt DAT on continually trained self-supervised models to deal with distortion-mismatch scenarios and evaluated the performance on different speech processing tasks of the Speech processing Universal PERformance Benchmark [1] (SUPERB). Experimental results showed that continually training a self-supervised model on unlabeled distorted data enhances the robustness of models, and DAT can further improve the robustness. The results showed that performing DAT with a continually trained self-supervised model not only improves performance on speech with distortions seen during training, but also on speech with new types of distortions that did not appear during training. In some cases, the proposed method even outperformed models trained directly on distorted labeled data. We also carried out the binary-domain and multi-domain setting with three different domain classifier objectives for DAT, and concluded that in some cases, it is suitable to treat all types of distorted speech as the same domain when we do not have domain knowledge of all speech data.

2. Noise-robust Self-supervised Learning

2.1. Problem formulation

In a typical SUPERB framework, an upstream model serves as a feature extractor $f(s;\theta_f)$. The input speech waveform s is input to the feature extractor to obtain speech representations $\hat{z}$. And a downstream speech processing model serves as a label predictor $y(\hat{z};\theta_y)$ to solve the downstream speech processing tasks. Domain mismatch occurs when the testing data has a different distribution compared to the training data. A common scenario is that speech distortions are involved during inference. This usually causes performance to degrade since the model may not

[⋆]The authors acknowledge the support of the 2022 Eighth Frederick Jelinek Memorial Summer Workshop.

have the ability to generalize to different types of distortions. In this work, we aim at adapting models θ_f to different domains to improve generalizability.

The problem formulation is stated as follows. Given a source domain dataset $\mathcal{S} = \{s_1, \cdots, s_N\}$ with N speech utterances and the corresponding ground truth labels $\mathbf{y} = \{y_1, \cdots, y_N\}$, one may train a model supervisedly with the source data and obtain good results on a testing set $\mathcal{C} = \{c_1, \cdots, c_L\}$ sampled from the same distribution. However, if a testing set $\mathcal{C}' = \{c'_1, \cdots, c'_L\}$ has speech distortions, the performance may suffer from degradation. To overcome this domain mismatch problem, assume there is an unlabeled dataset with K types of speech distortions $\mathcal{T} = \{\{\hat{s}_i^1\}, \{\hat{s}_i^2\} \cdots, \{\hat{s}_i^K\}\}$, where $\{\hat{s}_i^k\}$ is a set of speech utterances with distortions of type k. In this paper, we considered two scenarios: $\mathcal{C}' \in \mathcal{T}$ and $\mathcal{C}' \notin \mathcal{T}$. We utilized the unlabeled dataset $\mathcal{T}$ to update the self-supervised model θ_f in the hope that it can generalize well on the target domain testing set $\mathcal{C}'$. This can be performed during the upstream pre-training phase with continual training and during the finetuning phase with DAT.

2.2. Stage 1: upstream continual training

To adapt an upstream model to a new domain, [10, 11] performed continual training by training the pre-trained model with some additional steps on the data of the new domain. In our work, we continually train the upstream model with unlabeled data in $\mathcal{T}$ so that the upstream model can gain more model robustness at this stage.

2.3. Stage 2: domain adversarial training

To train the SUPERB framework in a domain adversarial manner, a domain classifier $d(\hat{z}; \theta_d)$ is introduced to predict the domain of the original speech input s. The parameters $\theta_f, \theta_y, \theta_d$ of the feature extractor, label predictor, and domain classifier are optimized according to the following equations:

$$\theta_y \leftarrow \theta_y - \alpha \frac{\partial L_y}{\partial \theta_y}, \qquad \theta_d \leftarrow \theta_d - \beta \frac{\partial L_d}{\partial \theta_d} \tag{1}$$

$$\theta_f \leftarrow \theta_f - \eta \left(\frac{\partial L_y}{\partial \theta_f} - \lambda \frac{\partial L_d}{\partial \theta_f} \right) \tag{2}$$

where L_f, L_y, and L_d are the losses of the feature extractor, label predictor, and domain classifier, respectively. α, β, and η are the learning rates for gradient descent. The feature extractor has a negative term of the domain classifier loss in its objective to ensure it is trained against the domain classifier to produce domain-indistinguishable representations. The gradient reversal parameter λ scales this negative loss term, and is a hyper parameter to be searched during training. For the objective of the domain classifier, there are different designs according to the two different settings elaborated in the following sections.

2.3.1. Binary-domain setting

Under the binary domain setting, speech data are categorized into only two domains: clean speech in $\mathcal{S}$ and speech with distortions added in $\mathcal{T}$. Under this setting, the domain classifier is optimized with the binary cross entropy objective to distinguish between clean and distorted speech. **Binary cross entropy** loss is denoted as $\mathcal{L}_{BCE}$:

$$\mathcal{L}_{BCE} = -\frac{1}{N+M} \sum_{i=1}^{N+M} d_i \cdot \log p_i + (1 - d_i) \cdot \log(1 - p_i) \tag{3}$$

where p_i is the output of the domain classifier after a sigmoid operation, and d_i is the domain class label for the i-th speech utterance. N and M are the quantity of the clean speech and the distorted speech examples, respectively. The goal of training adversarially with this objective is to obtain speech representations so that the domain classifier cannot distinguish the clean and noisy speech.

2.3.2. Multi-domain setting

Under the multi-domain setting, speech data are categorized into $K + 1$ domains, where the K kinds of distorted speech are viewed as K different target domains, and the clean speech forming the source domain. Two objectives, cross entropy loss and entropy loss, are involved under this setting.
Cross entropy loss is denoted as $\mathcal{L}_{CE}$:

$$\mathcal{L}_{CE} = -\frac{1}{N+M} \sum_{i=1}^{N+M} \sum_{k=0}^{K} d_i^k \log p_i^k \tag{4}$$

d_i^k is the k-th entry of the one-hot domain label vector of the i-th speech utterance. p_i^k is the k-th entry of the output of the domain classifier after softmax. In the process of DAT, maximizing the cross entropy loss results in misclassification of the domain classifier.
Entropy loss is denoted as $\mathcal{L}_E$ as shown below.

$$\mathcal{L}_E = -\frac{1}{N+M} \sum_{i=1}^{N+M} \sum_{k=0}^{K} p_i^k \log p_i^k \tag{5}$$

By maximizing this objective during DAT, the domain classifier is prone to output uniform probability distributions, in other words, making it output similar probability values for each domain class.

3. Experiment setups

3.1. Downstream speech processing tasks and models

In this work, we conducted experiments on five downstream speech processing tasks in the SUPERB benchmark [1]. **Intent Classification, IC** classifies the utterances into different intent classes. We use Fluent Command Speech [22] as our dataset, which has three intent classes: action, object, and location. The downstream model of this task is composed of a meanpooling operation followed by a linear layer. **Emotion Recognition, ER** predicts the speaker emotion from speech. We use IEMOCAP [23] as our dataset, which is composed of 151 videos and 9 emotions. The downstream model of this task is three layers of convolutional networks, one layer of self-attention, and two linear output layers. **Keyword Spotting, KS** detects some predefined keywords in the utterances. The Speech Commands v1.0 [24] is used as the dataset. The downstream model is composed of three linear layers. **Speaker Identification, SID** recognizes the speakers of the utterances. VoxCeleb1 [25] is used as the dataset. The downstream model follows the encoder-decoder architecture, where the encoder is a two-layer transformer followed by self-attention pooling, and the decoder being one linear layer. **Automatic Speech Recognition, ASR** transcribes the given utterances to text. LibriSpeech-clean-100 [26] is used as the training data, and LibriSpeech-dev-clean and LibriSpeech-test-clean are the validation set and testing set. The model architecture is two layers of bidirectional LSTM [27] with CTC [28] as the decoding strategy.

Table 1: *Evaluation results of* ***IC***, ***ER***, *and* ***KS*** *on the three testing sets of different distortion configurations. The best performances of the given tasks are marked bold.*

			IC (Acc)			**ER** (Acc)			**KS** (Acc)		
		continual	clean	m+g+r	fsd50k	clean	m+g+r	fsd50k	clean	m+g+r	fsd50k
(a)	baseline	-	99.47	96.94	97.47	63.96	57.33	60.55	97.14	93.38	93.80
(b)	oracle	-	99.55	99.34	98.21	70.41	69.31	69.31	97.57	96.46	95.26
(c)	w/o DAT	libri 100hr mgr	99.45	98.63	97.94	64.42	62.30	60.65	96.92	94.87	93.90
(d)	w/o DAT	libri 960hr mgr	99.39	98.84	97.89	67.28	67.47	65.62	97.12	96.11	94.77
			Cross Entropy Loss (CE)								
(e)	DAT $\lambda = 0.01$	-	99.47	98.68	97.47	68.85	63.59	63.50	**97.44**	95.26	94.64
(f)	DAT $\lambda = 0.001$	-	99.60	98.60	97.63	69.10	65.90	64.29	97.24	95.65	94.55
			Cross Entropy Loss (CE)								
(g)	DAT $\lambda = 0.001$	libri 100hr mgr	99.66	**99.45**	**98.55**	69.95	66.64	67.47	96.85	95.42	94.09
(h)	DAT $\lambda = 0.001$	libri 960hr mgr	99.55	99.39	98.31	**71.71**	**69.12**	**69.40**	97.05	**96.27**	**96.46**
			Entropy Loss (E)								
(i)	DAT $\lambda = 0.01$	-	99.55	98.47	97.36	63.87	59.91	59.26	96.92	94.94	94.06
(j)	DAT $\lambda = 0.001$	-	99.58	98.27	97.52	64.15	61.75	59.54	97.05	94.87	94.13
			Binary Cross Entropy Loss (BCE)								
(k)	DAT $\lambda = 0.01$	-	**99.68**	98.39	97.57	66.18	64.33	62.86	96.98	95.10	93.93
(l)	DAT $\lambda = 0.001$	-	99.60	98.97	97.89	68.76	64.52	64.52	97.27	95.59	93.90

3.2. Data configurations

We equally split the original training data of the downstream speech processing tasks into two splits, the clean split and the noisy split. We assume only the labels of the clean splits are available for training the downstream models. Because the labeled training data used in this paper are only half of the labeled data used in the original SUPERB paper [1], and the upstream models are not fixed during training, the performances in this paper are not comparable with the previous work. The noisy splits are distorted to serve as an unlabeled noisy dataset $\mathcal{T}$ in Section 2.1. Each speech utterance in the noisy splits contains one of the three distortion types, Musan [29] noise (m), Gaussian noise (g), and reverberation (r). The Musan noise corpus consists of various kinds of real-world noises, such as technical noise, domestic noise, traffic noise, etc. Gaussian noise is a statistical noise with a normal distribution as its probability density function. Reverberation is the phenomenon where sound or energy is preserved in a room or space due to reflection from obstacles. In the target domain split, among all the speech utterances, the three kinds of distortions, Musan noise, Gaussian noise, and reverberation, have the proportions 0.3, 0.4, and 0.3, respectively. Each utterance is only distorted with one kind of distortion. The signal-to-noise ratio (SNR) for additive noise is randomly sampled between 10 and 20 dB.

In this work, there are three testing set configurations. The first configuration is the original testing set for each downstream task provided by the SUPERB benchmark. The second configuration is constructed by adding the three kinds of distortions to the speech utterances in the testing split in the same way as $\mathcal{T}$. Thus, in the second configuration, the distortions are seen during training. The third configuration is meant to construct a new distortion at the testing time by adding noise utterances from the FSD50k [30] dataset. FSD50k is a subset of AudioSet [31]. It includes 200 sound classes mainly composed of real-world sound events. For **ASR**, we also tested the models on the CHiME3 [32] dataset, which includes real speech data recorded in 5 different locations.

3.3. Upstream model

In this work, we applied the HuBERT base model [33] pre-trained with 960 hours of Librispeech as the upstream model. For some experiments, we performed 60 epochs of continual training on the already pre-trained HuBERT base model with the LibriSpeech 100 hour or 960 hour split with three kinds of distortions (m, g, and r) added. For the data used for continual training, the proportions of different speech distortions are all 0.25 for Musan noise, Gaussian noise, reverberation, and clean speech.

For DAT, since the feature extractor needed to be trained adversarially against the domain classifier, we set the parameters of the upstream model trainable. For simplicity, only the last hidden states of the upstream model were used as the input of the downstream model.

3.4. Domain classifier

The domain classifier consists of a mean pooling procedure followed by a linear layer to project the features to the dimension of the number of total domains.

3.5. Training details

The Adam [34] optimizer is used for training. For all experiments, the learning rate η of the upstream model is set to $1\mathrm{e}{-5}$, since the upstream model is already pre-trained and requires smaller steps of parameter updates. For downstream models, the learning rate α of **KS** and **SID** are set to $1\mathrm{e}{-3}$ and $1\mathrm{e}{-1}$, respectively. For **IC**, **ER**, and **ASR**, the learning rate α of the downstream models are set to $1\mathrm{e}{-4}$. For the domain classifier, the learning rate β is set to $1\mathrm{e}{-4}$. For DAT, we searched the λ parameter in Eq. (2) across $1\mathrm{e}{-1}$ to $1\mathrm{e}{-4}$ in log scale and reported two of the best results for $\lambda = 1\mathrm{e}{-2}$ and $\lambda = 1\mathrm{e}{-3}$.

4. Results

Tables 1 and 2 are the results of the five downstream speech processing tasks on the testing sets with different distortion configurations. The baselines are trained supervisely with the

Table 2: *Evaluation results of **SID** and **ASR** on the three testing sets of different distortion configurations. For **ASR**, the results on CHiME3 are also reported.*

						ASR (WER)							
				SID (Acc)		clean		m+g+r		fsd50k		CHiME3	
		continual	clean	m+g+r	fsd50k	w/o	w/ LM	w/o	w/ LM	w/o	w/ LM	w/o	w/ LM
(a)	baseline	-	84.97	65.51	77.61	6.72	4.88	10.16	7.94	9.62	7.57	33.4	29.26
(b)	oracle	-	86.63	80.05	82.74	5.17	4.18	6.57	5.37	6.69	5.45	22.98	20.57
(c)	w/o DAT	libri 100hr mgr	87.02	70.91	80.96	6.23	4.87	8.04	6.47	7.90	6.38	27.82	24.27
(d)	w/o DAT	libri 960hr mgr	86.40	74.46	81.47	5.92	4.84	7.19	6.00	7.15	5.87	23.83	20.81
			Cross Entropy Loss (CE)										
(e)	DAT $\lambda = 0.01$	-	86.49	71.82	79.76	6.16	4.60	11.74	9.97	11.39	12.27	44.43	41.51
(f)	DAT $\lambda = 0.001$	-	87.44	72.28	79.94	6.29	5.77	9.70	9.00	9.67	8.68	32.98	29.03
			Cross Entropy Loss (CE)										
(g)	DAT $\lambda = 0.001$	libri 100hr mgr	88.70	79.59	83.57	5.75	4.82	7.30	6.21	7.21	6.15	25.60	22.73
(h)	DAT $\lambda = 0.001$	libri 960hr mgr	89.08	**80.27**	**85.04**	**5.49**	4.61	**6.82**	**5.69**	**6.67**	**5.62**	**23.44**	**20.71**
			Entropy Loss (E)										
(i)	DAT $\lambda = 0.01$	-	90.01	75.23	84.04	5.59	**4.45**	11.79	10.23	8.64	7.25	40.02	37.45
(j)	DAT $\lambda = 0.001$	-	90.50	73.43	84.46	6.01	4.59	9.97	7.74	8.88	7.31	32.77	28.90
			Binary Cross Entropy Loss (BCE)										
(k)	DAT $\lambda = 0.01$	-	89.25	73.71	82.12	6.23	4.68	9.68	7.56	9.18	7.38	31.20	28.72
(l)	DAT $\lambda = 0.001$	-	**90.96**	78.69	84.67	6.35	4.72	9.45	7.34	9.11	7.38	32.25	27.88

clean split training data. We also trained models with both the clean and noisy split data with the labels of the downstream task given. Since our proposed approaches do not use the labels of the noisy split, the above setting is considered as an oracle case (denoted as "oracle" in the tables).

4.1. Upstream continual training

Performing continual training on the pre-trained HuBERT base model with some unlabeled noisy data improves the performances of downstream tasks in most cases (rows (c),(d) vs (a)). Continual training on the noises m, g, and r not only improves the performance on the same distortion configuration but also improves the performance of unseen noise FSD50k and CHiME3 for **ASR**. However, for **KS**, the performance of the baseline on the clean testing set degrades when the upstream is continually trained. The same goes for the baseline of **IC**.

4.2. Domain adversarial training

Rows (e) and (f) show the results of DAT with cross entropy as the domain classifier objective (without continual training). By comparing DAT with the baseline results (rows (e)(f) vs (a)), DAT with cross entropy outperforms the corresponding baselines for tasks **IC**, **ER**, **KS** and **SID** in most cases. This suggests that DAT has the ability to solve speech processing tasks with different distortions and can generalize well on speech data with unseen distortions during the testing phase. It is also interesting to see that the two methods, DAT and continual training, improve different tasks under different configurations (rows (c)(d) vs (e)(f)). DAT does not degrade the clean speech performance for **KS** and **IC**. This shows that DAT can generalize to speech with distortions while preserving good performance on the clean testing set in most cases. However, DAT does not improve **ASR** that much compared to continual training.

Rows (g) and (h) are the results of DAT with continual training. DAT with continual training outperforms the baseline (rows (g)(h) vs (a)) and individual approaches (rows (g)(h) vs (c)(d)(e)(f)) in most cases. Note that for all of the speech tasks, there are cases where the performance of DAT with continual training exceeds the oracle in the unseen domain (rows (g)(h) vs (b)). This gives an insight into the potential of integrating continual training and DAT to better handle the domain shift problem when new domains are introduced.

4.3. Different objective functions

In this section, we compare the results of DAT using different objectives for training the domain classifier (rows (e)(f)(i)(j)(k)(l)). Under the multi-domain setting, for **ER** and **KS**, adopting cross entropy as the DAT objective yields the best performance in most of the cases. This suggests that for these two tasks, the feature extractor focuses on producing representations that cause the domain classifier to misclassify the domain but does not ensure uniform distributions for the domain classifier output. However, this is not the case for the results of **SID**. Applying DAT with entropy loss outperforms training with the cross entropy objective in every domain. This shows that the upstream model adversarially trained to maximize the entropy loss of the domain classifier is more suitable for SID. Under the binary-domain setting, DAT with binary cross entropy loss performs well for the **IC** and **SID** task. The implications of these results are that it is appropriate to view all kinds of distorted speech as the same domain, meaning that the goal of the upstream model is to output representations so that the domain classifier cannot tell whether the original waveform contains distortions or not. This is useful when there is no knowledge of the distortion types for the speech utterances during training.

5. Conclusions

DAT improves the robustness of self-supervised speech processing models. It leverages the unlabeled target domain data to generalize well in different domains. In the future, we will extend the training procedure to a multi-task scenario, where DAT can deal with different tasks and distortions simultaneously.

6. References

[1] S.-w. Yang, P.-H. Chi, Y.-S. Chuang, C.-I. J. Lai, K. Lakhotia, Y. Y. Lin, A. T. Liu, J. Shi, X. Chang, G.-T. Lin *et al.*, "SUPERB: Speech processing universal performance benchmark," *arXiv preprint arXiv:2105.01051*, 2021.

[2] M. Berouti, R. Schwartz, and J. Makhoul, "Enhancement of speech corrupted by acoustic noise," in *ICASSP'79. IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4. IEEE, 1979, pp. 208–211.

[3] J. Lim and A. Oppenheim, "All-pole modeling of degraded speech," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, no. 3, pp. 197–210, 1978.

[4] Y. Ephraim, "Statistical-model-based speech enhancement systems," *Proceedings of the IEEE*, vol. 80, no. 10, pp. 1526–1555, 1992.

[5] Y. Ephraim and H. L. Van Trees, "A signal subspace approach for speech enhancement," *IEEE Transactions on speech and audio processing*, vol. 3, no. 4, pp. 251–266, 1995.

[6] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 7–19, 2014.

[7] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, and Y. Bengio, "Multi-task self-supervised learning for robust speech recognition," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6989–6993.

[8] H. Wang, Y. Qian, X. Wang, Y. Wang, C. Wang, S. Liu, T. Yoshioka, J. Li, and D. Wang, "Improving noise robustness of contrastive speech representation learning with speech reconstruction," *arXiv preprint arXiv:2110.15430*, 2021.

[9] Q.-S. Zhu, J. Zhang, Z.-Q. Zhang, M.-H. Wu, X. Fang, and L.-R. Dai, "A noise-robust self-supervised pre-training model based speech representation learning for automatic speech recognition," *arXiv preprint arXiv:2201.08930*, 2022.

[10] Z. Chen, S. Wang, and Y. Qian, "Self-supervised learning based domain adaptation for robust speaker verification," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 5834–5838.

[11] W.-N. Hsu, A. Sriram, A. Baevski, T. Likhomanenko, Q. Xu, V. Pratap, J. Kahn, A. Lee, R. Collobert, G. Synnaeve *et al.*, "Robust wav2vec 2.0: Analyzing domain shift in self-supervised pretraining," *arXiv preprint arXiv:2104.01027*, 2021.

[12] 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.

[13] 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.

[14] H.-J. Na and J.-S. Park, "Accented speech recognition based on end-to-end domain adversarial training of neural networks," *Applied Sciences*, vol. 11, no. 18, p. 8412, 2021.

[15] 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.

[16] 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.

[17] S. Latif, J. Qadir, and M. Bilal, "Unsupervised adversarial domain adaptation for cross-lingual speech emotion recognition," in *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*. IEEE, 2019, pp. 732–737.

[18] D. Xin, Y. Saito, S. Takamichi, T. Koriyama, and H. Saruwatari, "Cross-lingual speaker adaptation using domain adaptation and speaker consistency loss for text-to-speech synthesis," *Proc. Interspeech 2021*, pp. 1614–1618, 2021.

[19] Q. Wang, W. Rao, S. Sun, L. Xie, E. S. Chng, and H. Li, "Unsupervised domain adaptation via domain adversarial training for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4889–4893.

[20] J. Huh, H. S. Heo, J. Kang, S. Watanabe, and J. S. Chung, "Augmentation adversarial training for self-supervised speaker recognition," *arXiv preprint arXiv:2007.12085*, 2020.

[21] C.-F. Liao, Y. Tsao, H.-Y. Lee, and H.-M. Wang, "Noise adaptive speech enhancement using domain adversarial training," *arXiv preprint arXiv:1807.07501*, 2018.

[22] L. Lugosch, M. Ravanelli, P. Ignoto, V. S. Tomar, and Y. Bengio, "Speech model pre-training for end-to-end spoken language understanding," *arXiv preprint arXiv:1904.03670*, 2019.

[23] C. Busso, M. Bulut, C.-C. Lee, A. Kazemzadeh, E. Mower, S. Kim, J. N. Chang, S. Lee, and S. S. Narayanan, "Iemocap: Interactive emotional dyadic motion capture database," *Language resources and evaluation*, vol. 42, no. 4, pp. 335–359, 2008.

[24] P. Warden, "Speech commands: A public dataset for single-word speech recognition," *Dataset available from http://download. tensorflow. org/data/speech_commands_v0*, vol. 1, 2017.

[25] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[26] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2015, pp. 5206–5210.

[27] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[28] 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.

[29] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[30] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, "Fsd50k: an open dataset of human-labeled sound events," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 829–852, 2022.

[31] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[32] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third 'chime'speech separation and recognition challenge: Dataset, task and baselines," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 504–511.

[33] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.