BOOSTING NOISE ROBUSTNESS OF ACOUSTIC MODEL VIA DEEP ADVERSARIAL TRAINING

Bin Liu[1,2] *Shuai Nie*[1,2,*] *Yaping Zhang*[1,2] *Dengfeng Ke*[1,3] *Shan Liang*[1] *Wenju Liu*[1]

[1] National Laboratory of Patten Recognition, Institute of Automation, Chinese Academy of Sciences, China
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences, China
[3] School of Information Science and Technology, Beijing Forestry University, China

{bin.liu2015,shuai.nie,yaping.zhang,dengfeng.ke,sliang,lwj}@nlpr.ia.ac.cn

ABSTRACT

In realistic environments, speech is usually interfered by various noise and reverberation, which dramatically degrades the performance of automatic speech recognition (ASR) systems. To alleviate this issue, the commonest way is to use a well-designed speech enhancement approach as the front-end of ASR. However, more complex pipelines, more computations and even higher hardware costs (microphone array) are additionally consumed for this kind of methods. In addition, speech enhancement would result in speech distortions and mismatches to training. In this paper, we propose an adversarial training method to directly boost noise robustness of acoustic model. Specifically, a jointly compositional scheme of generative adversarial net (GAN) and neural network-based acoustic model (AM) is used in the training phase. GAN is used to generate clean feature representations from noisy features by the guidance of a discriminator that tries to distinguish between the true clean signals and generated signals. The joint optimization of generator, discriminator and AM concentrates the strengths of both GAN and AM for speech recognition. Systematic experiments on CHiME-4 show that the proposed method significantly improves the noise robustness of AM and achieves the average relative error rate reduction of 23.38% and 11.54% on the development and test set, respectively.

Index Terms— robust speech recognition, deep adversarial training, acoustic model, generative adversarial net

1. INTRODUCTION

In realistic environments, speech is usually interfered by various noise and reverberation, which dramatically degrades the performance of automatic speech recognition (ASR) systems. Many neural network-based acoustic model training schemes have been proposed to boost the noise robustness of ASR, such as noise-aware training [1], noise adaptive training (NAT) [2] and DNN adaptation [3]. Various elaborate acoustic model architectures have also been proposed in order to promote the modeling capability. However, the complicated networks have high computational complexities and the degraded speech might lack its crucial discriminative characteristics, which makes it hard to significantly improve the ASR performance.

Another mainstream approach to boost noise robustness is adding a well-designed speech enhancement part during the front-end of ASR. The processing methods include traditional statistical methods like Wiener filter [4] and STSA-MMSE [5], a denoising autoencoder (DAE) [6] and DNN-based ideal binary masking [7]. However, more complex pipelines, more computations and even higher hardware costs (microphone array) are additionally consumed for this kind of methods. In addition, these enhancement methods use a particular form of loss functions such as mean squared error, which tends to generate over-smoothed spectra that lack the fine structures that are near to those of the true speech. The speech distortions and mismatches to training sometimes degrade the ASR performance. Moreover, it fails to optimize towards the final objective since the speech enhancement part is usually distinct from the recognition part, which leads to a suboptimal solution [8].

Generative adversarial nets (GANs) introduced by Goodfellow et al. [9] have attracted a lot of attention, which aims to generate samples following the underlying data distribution without the need for the explicit form of distribution. In addition, to obtain an optimal performance, a joint training of speech enhancement and acoustic model is proposed for robust speech recognition [10].

In this paper, we propose an adversarial training method to directly boost noise robustness of acoustic model. Specifically, a jointly compositional scheme of generative adversarial net (GAN) and neural network-based acoustic model (AM) is used in the training phase. GAN is used to generate clean feature representations from noisy features by the guidance of a discriminator that tries to distinguish between the true clean signals and generated signals. Without the complex front-end approaches and the need for one-to-one correspondence between the true clean and generated samples, our method greatly simplifies the training process. The use of adversarial training circumvents the limitation of hand-engineering loss functions and captures the underlying structural characteristics from the noisy signals, which could boost noise robustness of acoustic model.

This paper is organized as follows. The related work is discussed in Section 2. The basic and conditional generative adversarial nets are presented in Section 3 . The proposed deep adversarial training is described in Section 4. The experimental results and conclusions are given in Section 5 and Section 6, respectively.

2. RELATION TO PRIOR WORK

Generative adversarial nets (GANs) have attracted a lot of attention recently because of their successful applications in the computer vision and image processing tasks [11]. GANs have also been ap-

* denotes equal contribution to this work.

plied to speech conversion [12], synthesis [13] and enhancement [14] tasks. The use of GANs as classifiers has already been investigated for image classification tasks [15, 16], which extends GANs to learn discriminative classifiers by forcing the discriminator to output class labels. Shen et al.[17] used conditional generative adversarial nets as a classifier for the spoken language identification task. However, it's hard to achieve the equilibrium for a single discriminator network which plays two competing roles of fake samples identifier and labels predictor. Recently, Li et al. [18] proposed triple-GAN to introduce three components for both classification and class-conditional generation. To the best of our knowledge, using GANs for robust speech recognition has not yet been studied, so our method is the first approach to use the adversarial training framework for robust speech recognition.

3. GENERATIVE ADVERSARIAL NETWORKS

GANs are generative models introduced by Goodfellow et al [9], which consist of a generator (G) and a discriminator (D). The generator G produces samples from the data distribution $P(\mathbf{x})$ by transforming noise variables $\mathbf{z}$ into fake samples $G(\mathbf{z})$. The discriminator D is a classifier that aims to recognize whether the sample is from G or training data. G is trained to produce outputs that cannot be distinguished from "real" data by an adversarially trained D, which is trained to do as well as possible in detecting the generator's "fakes". More formally, this adversarial learning process is formulated as a two-player minimax game with the objective

$$\begin{aligned}\min_G \max_D V(G,D) = &\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}\left[\log D(\mathbf{x})\right] \\ &+ \mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z})}\left[\log\left(1 - D\left(G(\mathbf{z})\right)\right)\right].\end{aligned} \tag{1}$$

Regular GANs suffer from the vanishing gradients problem because of the sigmoid cross-entropy loss function adopted for the discriminator. The least-squares GANs (LSGANs) approach [19] substitutes the cross-entropy loss by the least-squares function, which can generate higher quality samples and perform more stable during the learning process. The objective functions for LSGANs can be defined as follows:

$$\begin{aligned}\min_D V_{\text{LSGAN}}(D) = &\frac{1}{2}\mathbb{E}_{\mathbf{x},\mathbf{x}_c\sim p_{\text{data}}(\mathbf{x},\mathbf{x}_c)}\left[\left(D(\mathbf{x},\mathbf{x}_c)-1\right)^2\right] \\ &+ \frac{1}{2}\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z}),\mathbf{x}_c\sim p_{\text{data}}(\mathbf{x}_c)}\left[\left(D(G(\mathbf{z},\mathbf{x}_c),\mathbf{x}_c)\right)^2\right] \\ \min_G V_{\text{LSGAN}}(G) = &\frac{1}{2}\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}(\mathbf{z}),\mathbf{x}_c\sim p_{\text{data}}(\mathbf{x}_c)}\left[\left(D(G(\mathbf{z},\mathbf{x}_c),\mathbf{x}_c)-1\right)^2\right].\end{aligned} \tag{2}$$

4. DEEP ADVERSARIAL TRAINING FOR ROBUST SPEECH RECOGNITION

For robust speech recognition, the main task of the acoustic model is to classify thousands of fine-grained senones given an input feature. We propose to train an acoustic model by the deep adversarial training method. The model consists of a generator(G), a discriminator (D) and a classifier(C), which is shown in the Fig. 1. In our case, the generator G performs the speech enhancement. It transforms the noisy speech signals into the enhanced version. The discriminator D aims to distinguish between the enhanced signals and clean ones. The classifier C classifies senones by features derivated from G. The

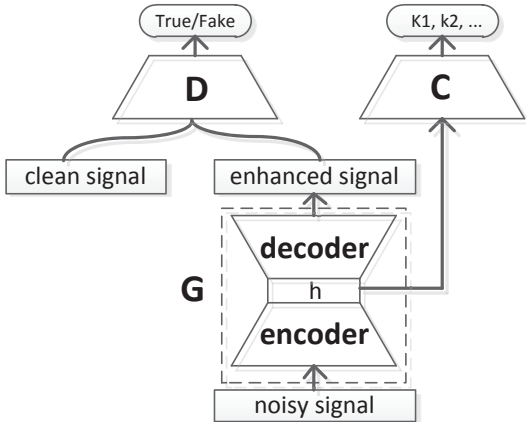

Fig. 1. The structure of proposed adversarial training method: G is used to do speech enhancement by transforming the noisy speech signals into the enhanced version, D aims to distinguish between the clean and the enhanced samples, and C is to recognize the sensones.

encoding parts of G and the classifier C can be regarded as an organic whole of acoustic model.

The generator G is an encoder-decoder, adapted from [14]. In the encoding stage, the input is passed and compressed through a series of strided convolutional layers followed by the leaky rectified linear units (LeakyReLUs) [20]. We choose the strided convolutions as they were shown to be more stable than other pooling approaches for GANs training [21]. Downsample is done until we get a condensed bottleneck representation, called the hidden vector $\mathbf{h}$. The encoding process is reversed in the decoding stage by means of transposed convolutions, followed again by LeakyReLUs.

For speech enhancement, there is a great deal of low-level information shared between the input and output, and it would be desirable to shuttle the information directly across the model. For example, the noisy and clean speech share the same underlying structure. Many low-level details could be lost to reconstruct the speech signal if we force all information to flow pass through the bottleneck layer. Therefore, we add skip connections following the general shape of a "U-Net" [22](Fig. 2). Specifically, each skip connection simply concatenates all channels at layer i with those at layer $n-i$, where n is the total number of layers. In addition, they are easier to optimize, as the gradients can flow deeper through the whole model [23].

The discriminator D is built to model the high frequencies of the data and it gets the true clean samples from the dataset and generated samples from the generator as input. Differing from other GAN works, it does not use $\mathbf{z}$. Isola et al.[24] report that adding a Gaussian noise as an input to G is not effective. We mention that there is no need for one-to-one correspondence between the true clean and generated samples.

The classifier C classifies senones and its input is the hidden vector $\mathbf{h}$ derivated from G. The bottleneck feature $\mathbf{h}$ is the compression of the noisy signal and captures the crucial structural characteristics with the guidance of D. Moreover, the classification progress is helpful to maintain more discriminative information for speech enhancement.

The G, D and C networks are jointly optimized by the adversarial training algorithm. Mathematically, given the noisy speech signal $\tilde{\mathbf{x}}$ and clean signal $\mathbf{x}$, G transforms it into enhanced version

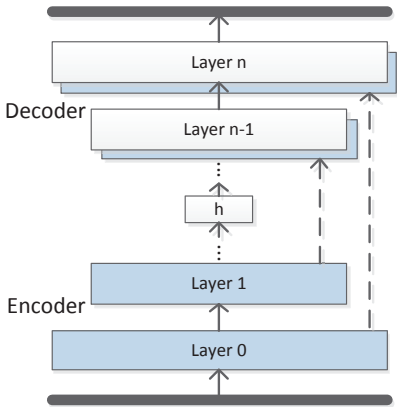

Fig. 2. "U-Net" architecture. It is an encoder-decoder with skip connections between mirrored layers in the encoder and decoder stacks. The dashed arrows denote skip connections.

$\hat{\mathbf{x}} = G(\tilde{\mathbf{x}}; \theta_g)$. D tries to classify a signal as the clean or enhanced. With the LSGANs approach(Eq. 2), the formulation is

$$\begin{aligned} \min_D V(D) = &\frac{1}{2}\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}\left[(D(\mathbf{x})-1)^2\right] \\ &+ \frac{1}{2}\mathbb{E}_{\tilde{\mathbf{x}}\sim p_{\text{data}}(\tilde{\mathbf{x}})}\left[(D(G(\tilde{\mathbf{x}})))^2\right]. \end{aligned} \tag{3}$$

The input of the classifier C is the hidden vector $\mathbf{h}$, which computes the posterior probabilities over HMM states. K classes labels $\{k_1, k_2, ...k_K\}$ correspond to the state-level forced alignment targets. C is optimized by minimizing the category loss

$$\min_C V(C) = \mathbb{E}_{\mathbf{h}\sim p(\mathbf{h})}\left[-\log(C(k|\mathbf{h}))\right]. \tag{4}$$

The generator G is trained to produce outputs that cannot be distinguished from "real" samples by the discriminator D. In this way, D is in charge of transmitting information to G of what is real and what is fake, such that G can correct its output towards the realistic distribution. The adversarial G loss, with the LSGANs approach(Eq. 2), becomes

$$\min_G V_{\text{GAN}}(G) = \frac{1}{2}\mathbb{E}_{\tilde{\mathbf{x}}\sim p_{\text{data}}(\tilde{\mathbf{x}})}\left[(D(G(\tilde{\mathbf{x}}))-1)^2\right]. \tag{5}$$

In addition, the classifier C can also guide the generator to learn more discriminative information in the reconstruction of the speech signals. Therefore, the final G loss is

$$\min_G V(G) = \alpha V_{\text{GAN}}(G) + V(C). \tag{6}$$

The magnitude of the adversarial loss is controlled by a hyper-parameter α. When $\alpha = 0$, the model is equivalent to a traditional deep neural network model.

As shown in algorithm 1, we alternatively train the parameters of D, G and C to fine-tune the model. Three components are implemented with neural networks and the parameters are updated by stochastic gradient descent.

Algorithm 1 The Deep Adversarial Training Algorithm

Input: Dataset $(\mathbf{x}, \tilde{\mathbf{x}}) \in (\mathbf{X}, \tilde{\mathbf{X}})$, $\mathbf{k}_i \in \mathbf{K}$, randomly initialized a generator G, a classifier C, a discriminator D parameterized by $\theta_g, \theta_c, \theta_d$, learning rate μ

Output: the optimized G, C and D parameterized by $\hat{\theta}_g, \hat{\theta}_c, \hat{\theta}_d$

// the adversarial training
repeat
  for number of training epochs **do**
    for number of mini-batches **do**
      // for discriminator
      $\theta_d \leftarrow \theta_d - \mu\frac{\partial V(D)}{\partial \theta_d}$
      // for generator
      $\theta_g \leftarrow \theta_g - \mu\frac{\partial V(G)}{\partial \theta_g}$
      // for classifier
      $\theta_c \leftarrow \theta_c - \mu\frac{\partial V(C)}{\partial \theta_c}$
    end for
  end for
until convergence
$\hat{\theta}_g = \theta_g, \hat{\theta}_c = \theta_c, \hat{\theta}_d = \theta_d$
return $\hat{\boldsymbol{\theta}}_{\boldsymbol{g}}, \hat{\boldsymbol{\theta}}_{\boldsymbol{c}}, \hat{\boldsymbol{\theta}}_{\boldsymbol{d}}$

5. EXPERIMENTS

5.1. Datasets

We systemically evaluate the proposed deep adversarial training method on the CHiME-4 corpus [25]. Two types of data are used in this paper: 'real data' that is recorded in real noisy environments (on a bus, cafe, pedestrian area, and street junction); 'simulated data' that has been generated by artificially mixing clean speech data from the WSJ0 set with noisy backgrounds. The training set consists of 1,600 real and 7,138 simulated utterances in the 4 noisy environments. Each utterance consists of six channels and we randomly choose one channel signals. The development and test sets consist of 3,280 and 2,640 utterances, respectively, each containing equal quantities of real and simulated data. We choose one channel signals according to the official instructions (dt/et05_simu/real_1ch_track.list). In addition, the original WSJ0 training data (si_tr_s, 7,138 utterances) is used as the clean speech for GANs training.

5.2. Setup

In following experiments, we take the 80-dimensional filter-banks as the input features, and each dimension of features is normalized to have zero mean and unit variance over the training set. To capture temporal information, 19 frames of context features are concatenated as the final inputs. Firstly, the splice context of noisy features $\tilde{\mathbf{x}}$ is feed into the generator G to generate its enhanced version $\hat{\mathbf{x}}$. Then the generated data used as 'False' samples and the randomly chosen true clean data used as 'True' samples form the input dataset of the discriminator D. We mention that there is no need for one-to-one correspondence between the true clean and generated samples. Finally, the classifier C uses the hidden output $\mathbf{h}$ of last encoder layer of G to compute the posterior probabilities of HMM state instead of G generating enhanced data. The frame-level targets required in training come from a well-trained Gaussian Mixture Model (GMM-HMM) system. In fact, the encoding parts of G and the classifier C can be regarded as an organic whole of acoustic model.

The generator G is composed of 8 2-D strided convolutional layers. Each convolutional block consists of a convolutional layer with 3×3 filters and 2×1 strides, followed by LeakyRelu function. As mentioned, the decoder stage of G is a mirroring of the encoder with the same filter widths and the same number of filters per-layer.

Table 1. WERs (%) of different hyper-parameter α on the development set

α	0.0	0.2	0.4	0.6	0.8
WER	18.96	17.19	16.32	16.66	16.96

Table 2. WERs (%) on the CHIME-4 corpus. 'Baseline' is the model following the Kaldi chime4/s5_1ch. 'CE-train' is the system trained by minimizing the cross-entropy loss. 'DA-train' is the proposed system optimized by deep adversarial training. Relative error rate reduction (PPER)(%) is also shown.

Data		Baseline	CE-train	DA-train	RERR
dev	simu	20.46	18.27	15.34	25.02
	real	22.13	19.65	17.30	21.83
	avg.	21.30	18.96	16.32	23.38
test	simu	29.94	27.66	24.75	17.33
	real	36.27	34.15	33.83	6.73
	avg.	33.11	30.91	29.29	11.54

However, skip connections make the number of feature maps in every decoder layer to be doubled. The classifier C is a multilayer perceptron (MLP) with two hidden layers of 1024 neurons. We use rectified linear unit (Relu) [26] as the activation of the hidden layer, and a softmax function of the output layer. The dropout [27] with a probability of 0.3 is added across the hidden layers. And we simply use a MLP with only one hidden layer as the discriminator D. The whole system is optimized by the adversarial training algorithm (denoted as 'DA-train'). We use the Adam optimizer [28] with a minibatch of size 256 and a learning rate of 0.0002. After finishing the training, we remove the D network and only use the combination of encoder layers of G and the classifier C for testing.

The baseline system is bulit following the Kaldi chime4/s5_1ch recipe [29](denoted as 'Baseline'). The acoustic model is a DNN with 7 hidden layers. After RBM pre-training, the model is trained by minimizing the cross-entropy loss. Note that we don't perform any sequence training or LM-rescoring methods because we focus on the frame level in this paper and will investigate methods at the full-sequence level in the future.

In addition, we also compare with the same acoustic model architecture as 'DA-train', but without the adversarial training. The comparison model is trained with same dataset and optimized by minimizing the cross-entropy loss (denoted as 'CE-train'). In fact, 'CE-train' is equivalent to the case of $\alpha = 0$ in Eq. 6.

The "CE-train" and "DA-train" systems are implemented with PyTorch [30]. The WSJ 5k trigram LM is used as the language model and the Kaldi WFST decoder for decoding in all the experiments.

5.3. Results

Firstly, we evaluate how the hyper-parameter α in Eq. 6 affects the performance of ASR. α specifies the tradeoff between the adversarial loss and the category loss for the optimization objective of G. When α is tiny, the category loss plays a main role and the adversarial loss rarely works, which may result in the acoustic model only focusing on discriminative feature and having worse generalization. On the other hand, when α is huge, the model may be lack of discriminant ability with weak constraint of category loss. Thus, an appropriate α is very important to get better performance of ASR. We explore the different α while keeping the other hyper-parameters fixed in the experiments. As shown in Table 1, the WER decreases with the increase of α. However, when α reaches 0.4, the WER will increase. Therefore, we choose $\alpha = 0.4$ for the following experiments.

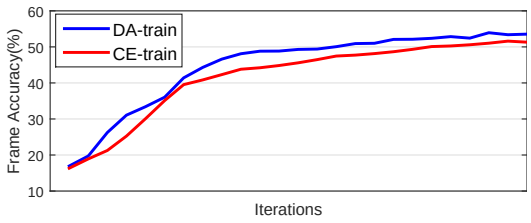

Fig. 3. Frame accuracy during training for DA-train and CE-train

Table 2 reports the WER results of different recognition models on the development and test dataset. In addition, we also report the relative error rate reduction (RERR) in the last column of Table 2. It can be seen that the proposed 'DA-train' consistently and significantly outperforms the baseline ('Baseline') and comparison ('CE-train') method on both development and test set. Compared with 'CE-train', the proposed 'DA-train' achieves significant performance improvements (from 18.96 to 16.32 on development set and from 30.91 to 29.29 on test set). It mainly owns to the proposed deep adversarial training boost noise robustness of acoustic model. Moreover, compared with 'Baseline', we achieve average RERRs of 23.38 % and 11.54% on the development and test set, respectively.

In order to further make a comparison between deep adversarial and cross-entropy training, we analyze the frame accuracy during training. Figure 3 shows that the model trained by adversarial manner not only arrives at higher frame accuracy but also learns more quickly. These results suggest that our proposed training method works efficiently and improves ASR performance.

6. CONCLUSIONS

In this paper, we propose an adversarial training method to directly boost noise robustness of acoustic model. Specifically, a jointly compositional scheme of generative adversarial net (GAN) and neural network-based acoustic model (AM) is used in training phase. GAN is used to generate clean feature representations from noisy features by the guidance of a discriminator that tries to distinguish between true clean signals and the generated signals. The joint optimization of generator, discriminator and AM concentrates the strengths of both GAN and AM for speech recognition. Systematic experiments on CHiME-4 show that the proposed method significantly improves the noise robustness of AM and achieves 23.38% and 11.54% relative improvements in WER. In the future, we will perform deep adversarial training for acoustic modeling at the full-sequence level and perform our experiments on a larger dataset.

7. ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China (No. 61573357, No. 61503382, No. 61403370, No. 61273267, No. 91120303).

8. REFERENCES

[1] Michael L Seltzer, Dong Yu, and Yongqiang Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7398–7402.

[2] Arun Narayanan and DeLiang Wang, "Joint noise adaptive training for robust automatic speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2504–2508.

[3] Hank Liao, "Speaker adaptation of context dependent deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7947–7951.

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

[5] Philipos C Loizou, "Speech enhancement: Theory and practice," *Crc Press*, 2013.

[6] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre Antoine Manzagol, "Extracting and composing robust features with denoising autoencoders," in *International Conference on Machine Learning*, 2008, pp. 1096–1103.

[7] Bo Li and Khe Chai Sim, "Improving robustness of deep neural networks via spectral masking for automatic speech recognition," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 279–284.

[8] Michael L. Seltzer, "Bridging the gap: Towards a unified framework for hands-free speech recognition using microphone arrays," in *Hands-Free Speech Communication and Microphone Arrays*, 2008, pp. 104–107.

[9] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *International Conference on Neural Information Processing Systems*, 2014, pp. 2672–2680.

[10] Tian Gao, Jun Du, Li-Rong Dai, and Chin-Hui Lee, "Joint training of front-end and back-end deep neural networks for robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4375–4379.

[11] David Berthelot, Tom Schumm, and Luke Metz, "Began: Boundary equilibrium generative adversarial networks," *arXiv preprint arXiv:1703.10717*, 2017.

[12] Chin-Cheng Hsu, Hsin-Te Hwang, Yi-Chiao Wu, Yu Tsao, and Hsin-Min Wang, "Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks," *arXiv preprint arXiv:1704.00849*, 2017.

[13] Bajibabu Bollepalli, Lauri Juvela, and Paavo Alku, "Generative adversarial network-based glottal waveform model for statistical parametric speech synthesis," in *INTERSPEECH*, 2017.

[14] Santiago Pascual, Antonio Bonafonte, and Joan Serr, "Segan: Speech enhancement generative adversarial network," 2017.

[15] Jost Tobias Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," *arXiv preprint arXiv:1511.06390*, 2015.

[16] Augustus Odena, "Semi-supervised learning with generative adversarial networks," 2016.

[17] Peng Shen, Xugang Lu, Sheng Li, and Hisashi Kawai, "Conditional generative adversarial nets classifier for spoken language identification," *Proc. Interspeech 2017*, pp. 2814–2818, 2017.

[18] Chongxuan Li, Kun Xu, Jun Zhu, and Bo Zhang, "Triple generative adversarial nets," 2017.

[19] Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K Lau, Zhen Wang, and Stephen Paul Smolley, "Least squares generative adversarial networks," 2016.

[20] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. ICML*, 2013, vol. 30.

[21] Alec Radford, Luke Metz, and Soumith Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *Computer Science*, 2015.

[22] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[24] Phillip Isola, Jun Yan Zhu, Tinghui Zhou, and Alexei A Efros, "Image-to-image translation with conditional adversarial networks," 2016.

[25] Emmanuel Vincent, Shinji Watanabe, Aditya Arie Nugraha, Jon Barker, and Ricard Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Computer Speech and Language*, 2016.

[26] Vinod Nair and Geoffrey E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *International Conference on International Conference on Machine Learning*, 2010, pp. 807–814.

[27] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting.," *Journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[28] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[29] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number EPFL-CONF-192584.

[30] "pytorch," `https://github.com/pytorch/pytorch`.