Adversarial Feature Learning and Unsupervised Clustering Based Speech Synthesis for Found Data With Acoustic and Textual Noise

Shan Yang, *Student Member, IEEE*, Yuxuan Wang, and Lei Xie, *Senior Member, IEEE*

***Abstract*—Attention-based sequence-to-sequence (seq2seq) speech synthesis has achieved extraordinary performance. But a studio-quality corpus with manual transcription is necessary to train such seq2seq systems. In this letter, we propose an approach to build high-quality and stable seq2seq based speech synthesis system using challenging found data, where training speech contains noisy interferences (acoustic noise) and texts are imperfect speech recognition transcripts (textual noise). To deal with text-side noise, we propose a VQVAE based heuristic method to compensate erroneous linguistic feature with phonetic information learned directly from speech. As for the speech-side noise, we propose to learn a noise-independent feature in the auto-regressive decoder through adversarial training and data augmentation, which does not need an extra speech enhancement model. Experiments show the effectiveness of the proposed approach in dealing with text-side and speech-side noise. Surpassing the denoising approach based on a state-of-the-art speech enhancement model, our system built on noisy found data can synthesize clean and high-quality speech with MOS close to the system built on the clean counterpart.**

***Index Terms*—Adversarial training, found data, sequence to sequence, speech synthesis.**

I. Introduction

RECENTLY, text-to-speech (TTS) has been significantly advanced with the wide use of deep neural networks (DNN). With the success of attention-based sequence-to-sequence (seq2seq) approach in machine translation [1], [2], DNN based speech synthesis has evolved into an *end-to-end* (E2E) framework, which unifies acoustic and duration modeling in a compact seq2seq paradigm, discarding frame-wise linguistic-acoustic mapping [3]–[7]. To achieve the best performance from a seq2seq system, studio-quality speech recordings with manual transcripts are necessary. Leveraging huge amount of speech resources available in public domain, or so-called *found data*, has drawn much interests lately. However, it is challenging to build a TTS system on low-quality found data as 1) speech may be contaminated by channel and environmental noises – *acoustic noise* and 2) transcripts generated by an automatic speech recognizer (ASR) contain inevitable errors–*textual noise*.

There are several recent studies addressing acoustic noise for E2E TTS. A straightforward idea is to use de-noised audio from speech enhancement to build the acoustic model [8]. But the inevitable distortion on training speech will propagate to the synthesized speech, resulting in clear quality deterioration. This conclusion has been further confirmed by another unsupervised source separation approach [9], where multi-node variational auto-encoder (VAE) was introduced to remove background music from the found speech for speech synthesis. The unstable separation directly affects the TTS quality. Another solution is to disentangle the speaker and noise attributes directly in the speech synthesis model. The approach in [10] first encodes the reference audio to disentangle speaker and noise with adversarial factorization, and then inject the encodings into the decoder to produce clean speech. But there is a strong assumption to conduct domain adversarial training– the audio of one speaker has a fixed type of acoustic noise. In more practical situation, recording conditions may vary and the collected data for the target speaker may come from different sources with different noise interferences.

We only find one recent study investigating the textual noise. In [11], the robustness of E2E systems to textual noise has been studied by manually corrupting text and using erronous ASR transcripts. Results suggest that E2E systems are only partially robust to training on imperfectly-transcribed data, and substitutions and deletions pose a serious problem. To the best our knowledge, there is still no solution to deal with textual noise in E2E TTS. Moreover, in many circumstances, building TTS systems on noisy found data has to deal with both textual and acoustic noise simultaneously – noisy speech transcribed by an ASR system with text errors.

This letter addresses both acoustic and textual noise interferences for building seq2seq-based speech synthesis system on noisy found data. To deal with textual noise, we propose a heuristic method to compensate the erroneous linguistic feature with phonetic information learned directly from speech. Specifically, VQVAE-based unsupervised clustering on the training speech is adopted to obtain latent phonetic representation, which is combined with the context vector from the text encoder output to produce synthesized speech. As for the acoustic noise, we propose to learn a noise-independent feature in the auto-regressive decoder through adversarial training and data augmentation,

Manuscript received April 22, 2020; revised September 6, 2020; accepted September 15, 2020. Date of publication September 21, 2020; date of current version October 8, 2020. This work was supported by the National Key Research and Development Program of China under Grant 2017YFB1002102. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Peter Jax *(Corresponding author: Lei Xie.)*

Shan Yang and Lei Xie are with the School of Computer Science, and Engineering, Northwestern Polytechnical University, Xi'an 710072, China (e-mail: syang@nwpu-aslp.org; lxie@nwpu.edu.cn).

Yuxuan Wang is with the Bytedance AI Lab, ByteDance, Mountain View, CA 94035 USA (e-mail: wangyuxuan.11@bytedance.com).

Digital Object Identifier 10.1109/LSP.2020.3025410

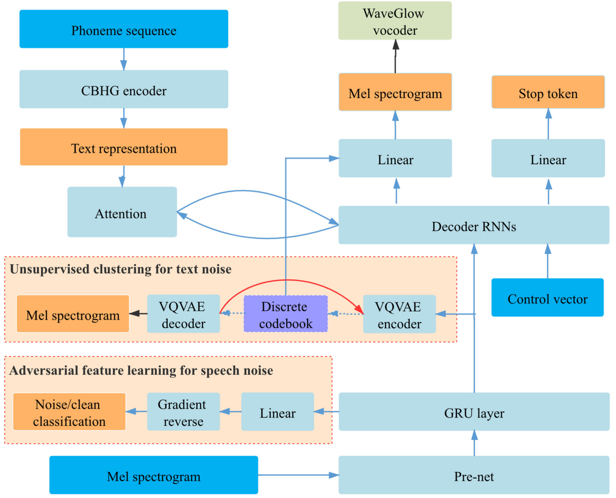

Fig. 1. Proposed methods for found data.

which does not need an extra speech enhancement model or strong assumption for noise conditions in [10]. Specifically, with the help of the clean data from another speaker, we adopt a domain classification network with a gradient reversal layer in the auto-regressive decoder to disentangle the noise conditions in latent feature space. Experiments show the effectiveness of the proposed approaches in dealing with textual and acoustic noise. Surpassing the denoising approach based on a state-of-the-art speech enhancement model, our system built on noisy data (speech SNR $=$ 4 dB, text CER=23.3%) can synthesize clean and high-quality speech with MOS close to the system built on the clean counterpart.

II. Proposed Methods for Found Data

Fig. 1 illustrates our seq2seq-based speech synthesis framework. It shares the similar architecture with Tacotron [3], [4], which is composed of a CBHG-based encoder [3] and an attention based auto-regressive decoder. The WaveGlow [12] vocoder is used to reconstruct the waveforms from Mel-spectrogram. Our approaches dealing with text and speech noise are built on this baseline system. Below we briefly describe the seq2seq-based speech synthesis.

For seq2seq TTS framework, suppose each speech utterance with M frames of acoustic features $\mathbf{y} = (y_1, y_2, \ldots, y_M)$ has corresponding N frames of golden character- or phoneme-level transcript $\mathbf{x} = (x_1, x_2, \ldots, x_N)$. The goal is to maximize the log probability $P(\mathbf{y}|\mathbf{x})$. And in the basic attention-based seq2seq framework [3], the decoder output $\hat{y}_t$ is computed from

$$\hat{y}_t = f(y_{t-1}, c_t) \quad \text{where } c_t = g(y_{t-1}, \mathbf{x}), \tag{1}$$

where y_{t-1} is the ground-truth acoustic frame at time $t-1$, and c_t is the context vector computed from the attention function $g(\cdot)$, which includes a content- or non-content based score function to measure the contribution of each x_i [1], [13], [14].

A. Unsupervised Clustering for Textual Noise

In the found data scenario, the golden transcript $\mathbf{x}$ is unavailable for model training. Thus we need an extra speech recognizer conducting auto-transcription to get $\hat{\mathbf{x}} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_K)$. Compared to the reference $\mathbf{x}$, $\hat{\mathbf{x}}$ may have irregular insertion, deletion or substitution errors. So the key problem is how to model the relations between unmatched speech $\mathbf{y}$ and text $\hat{\mathbf{x}}$. As described in Eq. (1), given a previous speech frame y_{t-1}, the attention mechanism computes the contribution of each $\hat{x}_i$ in hidden space to generate $\hat{y}_t$. Due to the recognition error of ASR and the monotonic nature of speech generation, the speech y_t may focus on the unrelated $\hat{x}_i$ rather than the correct x_i, which mostly causes the mispronunciation problem according to our experiments.

It's almost impossible to directly handle the text noise in the speech synthesis task as the supervision labels for text and speech are totally unavailable. Our approach dealing with text noise is motivated by recent works on unsupervised speech unit discovery, which has shown that phoneme-like clusterings can be automatically learned from speech in an unsupervised manner [15], [16]. Specifically, similar latent features from waveforms tend to be categorized into different clusterings that act as high-level speech descriptors closely related to phonemes [15]. To deal with text noise in found-data TTS, we propose a heuristic approach to conduct unsupervised clustering in the decoder to guide the speech generation, as shown in the unsupervised clustering module in Fig. 1.

In details, the context vector in the basic seq2seq framework is only computed from the output of text encoder, which inevitably contains text noise due to the inaccurate speech recognizer. In the proposed method, we compensate such errors with phonetic representation learned directly from speech. The context vector and the phonetic latent features are both injected to the decoder to produce synthesized speech, reducing the mismatch between speech and noisy text. There are several ways to learn the above discrete phonetic space [15], [17]. In our work, we adopt vector quantized variational autoencoder (VQVAE) to obtain a learnable discrete clusterings space e, which we assume is related to phoneme-like units [15].

Along with the basic auto-regressive process, the latent representation of y is also fed into the VQVAE encoder to obtain latent $z_e(y)$. We can obtain the discrete latent feature $z_q(y) \in e$ through

$$z_q(y) = e_k, \quad \text{where } k = \arg\min_i ||z_e(y) - e_i||_2. \tag{2}$$

Here $z_q(y)$ is treated as the latent phoneme representation clustered from speech and can be utilized to reconstruct back to speech through the VQVAE decoder.

Besides, the selected latent clustering $z_q(y)$ is also fed into the decoder along with the context vector c_t. During inference, the predicted previous frame $\hat{y}_{t-1}$ is utilized to obtain discrete representation. Therefore Eq. (1) is updated to

$$\hat{y}_t = f(y_{t-1}, c_t, z_q(y_{t-1})). \tag{3}$$

The objective function of the whole network is:

$$\begin{aligned} Loss = Loss_{recons} + ||sg[z_e(y)] - e||_2 \\ + \alpha * ||z_e(y) - sg[e]||_2 \end{aligned} \tag{4}$$

where $Loss_{recons}$ includes the reconstruction loss of both decoder and the VQVAE model, and $sg[\cdot]$ is a stop-gradient operator which has zero partial derivatives at the $\cdot$ operation. α is the weight of the commitment loss to make sure the encoder commits to an embedding [15]. Since there is no real gradient defined for Eq. (2), we copy the gradient from VQVAE decoder input to the encoder output, as shown in the red line in Fig. 1.

B. Adversarial Feature Learning for Acoustic Noise

Speech utterance $\mathbf{y}$ in found data may contain different types of background noise, which directly affects the performance of attention function $g(\cdot)$ and the whole model. We can apply an external speech enhancement module to obtain de-noised speech feature $\overline{\mathbf{y}}$ from $\mathbf{y}$ for downstream speech synthesis model training, but it may cause distortion problem in the generated speech [8], [9].

In order to mitigate the negative effects from speech noise in $\mathbf{y}$, we propose to use adversarial training to obtain the noise-independent latent feature $z_s = G_{adv}(\mathbf{y})$, where $G_{adv}(\cdot)$ is the proposed adversarial module. As shown in Fig. 1, the adversarial module contains a pre-net, a single unidirectional gated recurrent unit (GRU) network, and a classification network with a gradient reverse layer (GRL) [18]. The classification task is designed to classify the speech sample into clean/noisy. Here, since we do not have clean samples, similar to the data augmentation strategy in [10], we use another clean speech dataset along with the noisy samples to train the classification network. For a common classification network, the logistics of the last latent layer often represent the classification information (noise/clean condition in this work). When conducting the gradient reverse operation, its aim becomes disentangling the noise information to obtain the noise-independent features [18], or encouraging z_s not to be informative about the acoustic condition (noisy or clean). In [10], GRL is also adopted to disentangle noise from a reference audio to control the condition of speech synthesis. But we learn the noise-independent features z_s directly from the input speech $\mathbf{y}$. Therefore, in the speech synthesis stage, we do not need a clean reference audio to generate clean speech. With the GRL, the context vector c_t in Eq. (1) becomes

$$c_t = g(G_{adv}(y_{t-1}), \mathbf{x}). \tag{5}$$

Since there is an extra classification network, the final objective function is

$$Loss = Mel_{rmse} + \beta * Noise_{ce}, \tag{6}$$

where Mel_{rmse} denotes the Mel-spectrogram reconstruction loss, $Noise_{ce}$ is the cross entropy loss for noise classification, and β is the weight for the classification loss.

III. Experiments

In our experiments, we use an open-source Chinese corpus, which contains 10 hours speech of a female speaker. To obtain the target noisy dataset, we mix the clean speech with random types of noises from the CHiME-4 challenge [19]. We use a state-of-the-art chain model [20] based speech recognition module, trained with about 5000 hours speech, to transcribe the noisy speech, where the character error rate (CER) depends on the signal-to-noise ratio (SNR). In order to do the adversarial training, we use another clean corpus with 11 hours speech from another Chinese female speaker as the *clean* data. Another copy of this corpus is mixed with random noise from CHiME-4, together with the target noisy corpus above to form the *noisy* data. We use an internal speech recognition system to obtain transcripts. The CER is 8.9% for the clean target speaker. As for the speech enhancement baselines, we test an unsupervised model Separabl [9] and a state-of-the-art supervised model DCUnet [21]. The Separabl model is trained with the noisy target data only, and the DCUnet is trained with the speech and noise data from Deep Noise Suppression Challenge [22]. The Perceptual Evaluation of Speech Quality (PESQ) for Separabl and DCUnet are 2.02 and 3.00, respectively.

We mainly analyze the phone, tone and prosody information through our text analysis module to obtain text representation. For the speech representation, 80-band mel-scale spectrogram is treated as y for the decoder. We reserve 400 sentences from the target corpus for evaluation. There are 20 normal-hearing listeners aged from 20 to 25 attending the mean opinion score (MOS) test as subjective evaluation,[1] where the MOS ranges from 0.5 to 5 at every 0.5 for chosen. Since the length of the predicted feature is different from the target one, we conduct dynamic time warping to align the two sequences and compute the mel-cepstral distortion (MCD) for objective evaluation.

A. Model Details

1) Basic Architecture: For the basic seq2seq system, we follow the architecture of Tacotron and Tacotron2 [3], [4]. In the encoder, we adopt three feed-forward layers as pre-net followed by a CBHG module [3]. As for the decoder, the acoustic feature y is firstly fed into the decoder pre-net. And a unidirectional LSTM with GMM based attention mechanism [14] is adopted on the latent features. The basic architecture is applied to the baseline and topline systems.

2) Unsupervised Clustering: In the proposed unsupervised clustering for dealing with textual noise, the output of decoder pre-net is fed into the VQVAE encoder, which contains two layers feed-forward networks with 256 units followed by ReLu activation. The VQVAE decoder shares the similar architecture with the above encoder. For the vector quantization module, there are 256 code vectors with 128 dimensions in the code book. The weight of commit loss is set to 0.25.

3) Adversarial Feature Learning: For the noise-independent feature learning to deal with the acoustic noise, we add a 256-unit unidirectional GRU layer on the top of pre-net in the decoder. The output of GRU layer is fed into the following decoder LSTM layer with controllable speaker and noise condition, as well as the classification network.

B. Experimental Results

1) Basic Systems: We first evaluate the effects of textual and acoustic noise in training data on the baseline systems. Table I shows the objective and subjective results, where CGER means the character-level generation error. There are 7338 Chinese characters in total in the 400 test sentences.

System A is the topline trained using golden transcripts and clean speech. We use System B to E to examine the textual noise only, while using clean speech data to train the model. Compared to the topline, System B to D get worse as the CER increases in both objective and subjective tests, which confirms the negative effects of textual noise. Comparing System A, B and C, although the text noise affects both objective and subjective results, we find the seq2seq based model has few pronunciation errors when

[1] Samples can be found at https://syang1993.github.io/found-data

TABLE I
THE PERFORMANCE OF BASIC ARCHITECTURE FOR DIFFERENT TYPES OF FOUND DATA, WHERE R MEANS RECORDINGS

Index	CER (%)	SNR	ATT	MOS	MCD	CGER (%)
R	-	-	-	4.41	-	-
A	0	clean	GMM	4.21	3.08	0.05
B	8.8	clean	GMM	3.62	4.29	0.07
C	11.7	clean	GMM	3.51	4.34	0.10
D	23.3	clean	GMM	3.04	4.55	3.24
E	23.3	clean	LSA	2.63	4.63	9.69
F	0	8 dB	GMM	2.10	7.16	0.05
G	0	4 dB	GMM	1.79	8.78	0.04
H	23.3	4 dB	GMM	0.78	-	-

TABLE II
THE PERFORMANCE FOR INDIVIDUAL TEXTUAL NOISE

Index	CER (%)	MOS	MCD	CGER (%)
A	0	4.21	3.08	0.05
VQVAE_A	0	4.25	3.10	0.06
D	23.3	3.04	4.55	3.24
VQVAE_D	23.3	3.47	4.42	1.29

CER$< 10\%$. Since there is no punctuation in the noisy ASR transcription, the prosody of generated speech of system B and C is unsatisfactory, which causes the worse MOS values. This can be further solved through a more robust prosody model. For more noisy text, the generated speech of System D suffers from the mispronunciation problem. Since the context vector c_t directly depends on the attention alignment, we also compare the content and non-content score function. In System E, we conduct content-based location sensitive attention (LSA) [13], where text memory is taken into account. System D with non-content-based GMM attention outperforms the LSA-based System E. This is because noisy text is used to obtain the alignments in LSA, which affects the attention accuracy.

As for the acoustic noise, we use the golden transcripts to build System F and G with noisy training speech at 8 dB and 4 dB SNR, respectively. It's obvious that System F outperforms System G since the speech data used in F contains less noise. But the synthesized speech of both systems is noisy. System H represents the real found data condition without correct transcription and clean speech, which achieves the lowest MOS of 0.78. Note that we do not have MCD for system H, since it always generates noise and even cannot stop during generation due to the system failure caused by model trained with noisy speech.

2) Unsupervised Clustering: To overcome the textual noise, we then build a system with proposed unsupervised clustering to mitigate the mispronunciation problem caused by wrong transcriptions. Table II shows the performance, where VQVAE based unsupervised clustering is used in the topline System A and baseline System D. Table II shows that the proposed VQVAE_D significantly outperforms System D in both subjective and objective metrics. The proposed method decreases the character generation error rate from 3.24% to 1.29%. It's because that each output $\hat{y}_t$ also depends on the unsupervisedly discovered units from speech, which can mitigate the textual noise during training. Besides, comparing System A with VQVAE_A, we find that unsupervised clustering will not degrade the performance of the topline system.

TABLE III
THE PERFORMANCE FOR TEXTUAL AND ACOUSTIC NOISE

Index	CER (%)	SNR	MOS	MCD	CGER
H	23.3	4 dB	0.78	-	-
Separabl [9]			2.35	8.70	3.67%
DCUnet [21]			3.23	5.32	2.11%
Adv-sen			3.50	6.51	0.88%
Adv-frame			4.05	5.03	0.23%

3) Adversarial Feature Clustering: In real applications, we may have to deal with both acoustic and textual noises. Here we use adversarial feature clustering to improve System H. The performances of different approaches are summarized in Table III. We can see that the two systems that use speech enhancement to remove noises before TTS model training can obviously improve TTS performance. In System Separabl, we do not need external data for speech enhancement model training, where the de-noised speech is directly obtained through the pre-trained unsupervised multi-node VAE model from noisy data [9]. For System DCUnet, we need extra multi-speaker speech data and noise data to train the speech enhancement model. We notice that the proposed adversarial feature learning method significantly outperforms the speech enhancement methods. System Separabl indeed decreases the noise interference in the generated speech, but the performance of such unsupervised speech enhancement is not stable, which causes obvious speech distortions in the synthesized speech. Although System DCUnet, which adopts supervised speech enhancement, shows better ability of de-noising, it also suffers a lot from mispronunciation errors. Besides, there are also some noticable distortions in the generated speech. Note that for the proposed adversarial feature clustering method, we need extra clean speech data from another speaker as augmentation, but speech enhancement model is not needed.

With the help of another clean speech synthesis dataset, we conduct adversarial training to obtain noise-independent features in both sentence- and frame-level, named Adv-sen and Adv-frame respectively. For System Adv-sen, classification is conducted on the mean and variance of latent features to obtain sentence-level representation. We find although it can produce clean speech with good prosody, generated speech is not stable on pronunciations. With frame-level adversarial feature learning, system Adv-frame achieves the best performance among all systems. We assume the result benefits from two aspects: 1) the auxiliary dataset decreases the CER in the whole training texts and guides the model how to generate clean speech, 2) the adversarial feature learning can disentangle the noise information from speech, hence the control vector can directly control the generation process for producing clean speech.

IV. CONCLUSIONS AND FUTURE WORK

This letter proposes an unsupervised clustering method to handle textual noise in found data, and an adversarial feature learning method to generate clean synthesized speech with noisy training speech. Experiment shows that the proposed methods are effective to build high-quality and stable seq2seq based speech synthesis model for noisy found data. Future work will explore more robust methods to handle textual noise and test our approach on real found data. Room reverberation is also another interference that desires further investigation.

REFERENCES

[1] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. Int. Conf. Learn. Representations*, 2015.

[2] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Conf. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.

[3] Y. Wang *et al.*, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, 2017, pp. 4006–4010.

[4] J. Shen *et al.*, "Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 4779–4783.

[5] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2018, pp. 4784–4788.

[6] N. Li, S. Liu, Y. Liu, S. Zhao, M. Liu, and M. Zhou, "Neural speech synthesis with transformer network," in *Proc. Assoc. Adv. Artif. Intell.*, 2019, pp. 6706–6713.

[7] S. Yang *et al.*, "On the localness modeling for the self-attention based end-to-end speech synthesis," *Neural Netw.*, vol. 125, pp. 121–130, 2020.

[8] C. Valentini-Botinhao, X. Wang, S. Takaki, and J. Yamagishi, "Investigating RNN-based speech enhancement methods for noise-robust text-to-speech," in *Proc. Speech Synthesis Workshop*, 2016, pp. 146–152.

[9] N. Gurunath, S. K. Rallabandi, and A. Black, "Disentangling speech and non-speech components for building robust acoustic models from found data," 2019, *arXiv:1909.11727*.

[10] W.-N. Hsu *et al.*, "Disentangling correlated speaker and noise for speech synthesis via data augmentation and adversarial factorization," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 5901–5905.

[11] J. Fong, P. O. Gallegos, Z. Hodari, and S. King, "Investigating the robustness of sequence-to-sequence text-to-speech models to imperfectly-transcribed training data," in *Proc. Interspeech*, 2019, pp. 1546–1550.

[12] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *IEEE Int. Conf. Acoust., Speech Signal Process.*, 2019, pp. 3617–3621.

[13] J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Proc. Nat. Poisons Inf. Service*, 2015, pp. 577–585.

[14] E. Battenberg *et al.*, "Location-relative attention mechanisms for robust long-form speech synthesis," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, 2020, pp. 6194–6198.

[15] A. van den Oord *et al.*, "Neural discrete representation learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6306–6315.

[16] E. Dunbar *et al.*, "The zero resource speech challenge 2019: TTS without T," in *Proc. Interspeech*, 2019, pp. 1088–1092.

[17] N. Dilokthanakul *et al.*, "Deep unsupervised clustering with Gaussian mixture variational autoencoders," 2016, *arXiv:1611.02648*.

[18] S. Sun, B. Zhang, L. Xie, and Y. Zhang, "An unsupervised deep domain adaptation approach for robust speech recognition," *Neurocomputing*, vol. 257, pp. 79–87, 2017.

[19] E. Vincent, S. Watanabe, A. A. Nugraha, J. Barker, and R. Marxer, "An analysis of environment, microphone and data simulation mismatches in robust speech recognition," *Comput. Speech Lang.*, vol. 46, pp. 535–557, 2017.

[20] D. Povey *et al.*, "Purely sequence-trained neural networks for ASR based on lattice-free MMI," in *Proc. Interspeech*, 2016, pp. 2751–2755.

[21] H.-S. Choi, J.-H. Kim, J. Huh, A. Kim, J.-W. Ha, and K. Lee, "Phase-aware speech enhancement with deep complex u-Net," in *Proc. Int. Conf. Learn. Representations*, 2019.

[22] C. K. Reddy *et al.*, "The interspeech 2020 deep noise suppression challenge: Datasets, subjective testing framework, and challenge results," 2020, *arXiv:2005.13981*.