A TEACHER-STUDENT FRAMEWORK FOR UNSUPERVISED SPEECH ENHANCEMENT USING NOISE REMIXING TRAINING AND TWO-STAGE INFEERENCE

Li-Wei Chen², Yao-Fei Cheng², Hung-Shin Lee¹, Yu Tsao³, and Hsin-Min Wang²

¹North Co., Ltd., Taiwan
²Institute of Information Science, Academia Sinica
³Research Center for Information Technology Innovation, Academia Sinica

ABSTRACT
The lack of clean speech is a practical challenge to the development of speech enhancement systems, which means that the training of neural network models must be done in an unsupervised manner, and there is an inevitable mismatch between their training criterion and evaluation metric. In response to this unfavorable situation, we propose a teacher-student training strategy that does not require any subjective/objective speech quality metrics as learning reference by improving the previously proposed noisy-target training (NyTT). Because homogeneity between in-domain noise and extraneous noise is the key to the effectiveness of NyTT, we train various student models by remixing the teacher model’s estimated speech and noise for clean-target training or raw noisy speech and the teacher model’s estimated noise for noisy-target training. We use the NyTT model as the initial teacher model. Experimental results show that our proposed method outperforms several baselines, especially with two-stage inference, where clean speech is derived successively through the bootstrap model and the final student model.

Index Terms—speech enhancement, noise remixing

1. INTRODUCTION
Speech Enhancement (SE) aims to improve audio quality by removing noise from speech signals. It has a wide-range of applications, such as the front-end of automatic speech/speaker recognition systems [1, 2], where the SE module removes noise from noisy inputs, thereby improving recognition results. The success of current SE development mainly relies on training data containing a large number of pairs of clean and noisy speech [3, 4, 5, 6, 7, 8, 9, 10, 11]. During training, noisy speech is usually synthesized by mixing clean speech and noise. However, due to the high cost of recording, it is difficult to collect clean speech and in-domain noise in real-world scenarios.

Many methods that operate in an unsupervised manner have recently been proposed to address this problem [3, 4, 5, 6, 7, 8, 9, 10, 11]. Generally, unsupervised learning in SE can be defined from a negative perspective: the use of paired/parallel noisy and clean speech during training is prohibited or infeasible [14]. Fu et al. divided this definition into three levels: 1) clean speech or in-domain noise is required; 2) noisy speech is required; and 3) no training data is required [17]. Under this definition, the noisy-target training (NyTT) method proposed by Fujimura et al. can be regarded as employing a level-1 training strategy. It trains the SE model with original noisy speech as the training target. The original noisy speech is used to mix with extraneous noise to form noisier input speech that needs to be enhanced. Despite NyTT’s competitive performance, it shares a disadvantage with SE models trained with paired clean and noisy data. That is, NyTT only performs well when the extraneous noise is close to the in-domain noise contained in the test noisy speech. If the extraneous noise is not similar to the in-domain noise, the out-of-domain (OOD) issue can easily distract the processing power of the NyTT model, because it has to deal with different noise than the noise seen in training.

To overcome the OOD problem, unsupervised algorithms have been proposed. For example, mixture invariant training (MixIT) in speech separation enables unsupervised domain adaptation and learning from large amounts of real-world data without the need for ground-truth source waveforms [18]. Although MixIT has been successfully adapted to various SE tasks, it requires access to the in-domain noise. To address this issue, Tzinis et al. proposed RemixIT [19], which adopts a teacher-student training framework to achieve state-of-the-art (SoTA) performance on various unsupervised and semi-supervised SE tasks. The flexibility of the framework allows the use of any SE model as the teacher model. Inspired by NyTT and RemixIT, we propose a level-2 unsupervised SE model based on the teacher-student framework.

It is known that when the training data used for an SE model matches the test data, the performance of the test is higher, and vice versa. This makes domain matching an important performance factor. This is especially important in real-world scenarios where the noise is complex and it is difficult to synthesize similar noise during model training. Therefore, we believe that this issue needs to be addressed urgently. Our contributions include: 1) solving the domain mismatch...
problem; 2) eliminating the need for in-domain data other
than noisy speech; and 3) proposing a new model suitable for
two-stage inference so that the model performance of NyTT
can be further improved. Additionally, we open-sourced the
code for training on GitHub.

2. RELATED WORK

2.1. Noisy-target Training (NyTT)

Traditionally, supervised SE methods use clean speech as the
training target, and the noisy input speech is synthesized by
mixing clean speech and noise. Fujimura et al. called this
approach clean target training (CTT). Unlike CTT, in NyTT
[13], the clean target speech is replaced by noisy speech. As
shown in Fig. 1 (the blue part), the i-th sample y_i in a training
batch of the noisy input speech Y ∈ ℝ^{B×M} is synthesized by
mixing the noisy speech x_i in X ∈ ℝ^{B×M} and the noise n_i in
N ∈ ℝ^{B×M}, where B is the batch size, and M is the
signal length. The input Y is then fed into the model to get
the estimated result Š^T ∈ ℝ^{B×M}. NyTT uses mean squared
error as the loss function to update the model:

\[ \mathcal{L}_{NyTT} = \frac{1}{B} \sum_{i=1}^{B} \| x_i - Š^T \|_2^2. \]  

(1)

2.2. RemixIT

RemixIT is a teacher-student training framework with SoTA
results on several unsupervised and semi-supervised denoising
tasks by adapting a model’s domain to another domain
[19]. It uses the speech and noise estimated by the teacher
model to form paired training data for student model training.
Specifically, the model trained with OOD data is used as
the initial teacher model θ^T_0. During training, the teacher
model θ^T_k estimates the speech Š^T ∈ ℝ^{B×M} and noise Ń^in ∈
ℝ^{B×M} from the in-domain noisy speech X ∈ ℝ^{B×M}:

\[ (Š^T, Ń^in) = θ^T_k(X). \]  

(2)

Then, the new noisy speech Y ∈ ℝ^{B×M} for training the stu-
dent model is synthesized by mixing the estimated speech Š^T
and the shuffled estimated noise Ń^in = (n^i_1)_{i=1}^B = P Ń^in ∈
ℝ^{B×M}. P is a B × B permutation matrix used to generate
random-order in-domain noise from Ń^in.

The teacher model θ^T_k is updated according to one of the
following two Teacher Update Protocols (TUPs):

- **Static teacher**: The teacher model is not updated during
  the training of the student model.
- **Exponentially moving average teacher**: For each epoch,
  the teacher model is replaced by the weighted sum of the
  latest student model and the current teacher model, i.e.,
  \( θ^T_{k+1} = γ θ^S_k + (1 − γ) θ^T_k \), where \( γ = 0.005 \).

![Fig. 1. An overview of the proposed method. The blue dashed
line only exists when Bootstrap θ^T_{\text{CTT}}, i.e., the initial teacher
model, is trained by NyTT. The gray dashed line shows the
flow how the predicted in-domain noise is obtained from the
k-th inference-in-training by Teacher θ^T_{\text{CTT}}.]

3. PROPOSED METHOD

3.1. Training Framework

We use NyTT to train the initial teacher model. The blue
lines in Fig. 1 illustrate the training process of the initial
teacher model θ^T_0 and the inference-in-training process of the
teacher model θ^T_k. The noisy speech X and the extraneous
noise N^ext ∈ ℝ^{B×M} are mixed into the noisy input speech
Y. Different from the loss function in Eq. 4 referring to the
DEMUCS architecture [20], the model is updated by mini-

mizing the mean absolute error of the noisy speech X and the
estimated noisy speech Š^T.

Given the initial teacher model θ^T_0, the epoch size E, the
number of mini-batch B_{\text{mini}}, the batch size B, and TUP,
we follow the teacher-student training process of RemixIT.
In each epoch, we first sample a batch of noisy speech X and
initialize a random B × B permutation matrix for shuffling
the estimated in-domain noise Ń^in. We then estimate speech
and in-domain noise from X and remix in-domain noisy speech
for student model training. After the student model is trained,
the teacher model is updated according to the given TUP.

We propose six methods to train the student model. Three
of them are based on CTT, and the others are based on NyTT.
The CTT-based methods are shown on the left side of Fig. 1
(θ^T_{\text{CC}}: the red part), while the NyTT-based methods are shown
on the right side of Fig. 1 (θ^T_{\text{SN}}: the orange part). Note that
n^mix in Fig. 1 can be a sample of in-domain noise, a sample
of in-domain noise or extraneous noise, or a mixed sample of
in-domain noise and extraneous noise. The six student model
training methods are described as follows:

- **CTT-1** takes noisy speech X as input (i.e., Y = X) and the
  speech estimated by the teacher model Š^T as the target.
- **CTT-2** takes the remix of estimated speech Š^T and noise
  }
\(N^{mix}\) as input (i.e., \(Y = \hat{S}^T + N^{mix}\)) and \(\hat{S}^T\) as the target. \(N^{mix}\) contains only in-domain noise (i.e., \(N^{mix} = N^{in}\)).

- **CTT-3** takes the remix of estimated speech \(\hat{S}^T\) and noise \(N^{mix}\) as input (i.e., \(Y = \hat{S}^T + N^{mix}\)) and \(\hat{S}^T\) as the target. \(N^{mix}\) is a mixture of in-domain and extraneous noise (i.e., \(N^{mix} = N^{in} + N^{ext}\)).

- **NyTT-1** takes the remix of noisy speech \(X\) and noise \(N^{mix}\) as input (i.e., \(Y = X + N^{mix}\)) and \(X\) as the target. \(N^{mix}\) contains only in-domain noise (i.e., \(N^{mix} = N^{in}\)). Its complete training process is summarized in Algorithm 1.

- **NyTT-2** takes the remix of noisy speech \(X\) and noise \(N^{mix}\) as input (i.e., \(Y = X + N^{mix}\)) and \(X\) as the target. Each sample in \(N^{mix}\) is from \(N^{in}\) or \(N^{ext}\).

- **NyTT-3** takes the remix of noisy speech \(X\) and noise \(N^{mix}\) as input (i.e., \(Y = X + N^{mix}\)) and \(X\) as the target. \(N^{mix}\) is a mixture of in-domain and extraneous noise (i.e., \(N^{mix} = N^{in} + N^{ext}\)).

### 3.2. Two-stage Inference

As mentioned in Section 1, the enhanced speech \(\hat{S}^T\) may still contain some in-domain noise that cannot be completely removed by the initial NyTT model \(\theta_{T}^0\). Although Grzywalski et al. claim that performing speech enhancement through the same network up to five times improves speech intelligibility [21], there is no significant improvement when we pass noisy speech through \(\theta_{T}^0\) twice (see Table 1). Instead of using the same model in multi-stage inference, we successively use the initial teacher model \(\theta_{T}^0\) and the final model (i.e., \(\theta_{S}^k\) or \(\gamma \theta_{S}^k + (1 - \gamma) \theta_{T}^k\)) for two-stage inference. In the first stage, \(\theta_{T}^0\) is used to enhance noisy speech. Then, we feed the enhanced speech into \(\theta_{S}^k\) or \(\gamma \theta_{S}^k + (1 - \gamma) \theta_{T}^k\) in the second stage. We were surprised to see a considerable improvement in this practice.

### 4. EXPERIMENTS

#### 4.1. Datasets

We used VoiceBank-DEMAND as the in-domain noisy speech dataset [22]. It is part of the training set in the original NyTT study [13]. The training set consists of 28 speakers (11,572 utterances) with four signal-to-noise ratios (SNR: 15, 10, 5, and 0 dB). The test set consists of two speakers (824 utterances) with four SNRs (17.5, 12.5, 7.5, and 2.5 dB). We also used the CHiME-3 backgrounds as the extraneous (OOD) noise set [23]. It is also part of the training noise set in the original NyTT study.

#### 4.2. Model Structure

We used DEMUCS as the model architecture [20]. It was developed for real-time SE in the waveform domain and has been widely adopted in academia and industry. It consists of a UNet connected encoder and decoder, and the configurable parameters are the number of layers (\(L\)) and the number of initial hidden channels (\(H\)). We upsampled the input audio by the resampling factor \(U\), fed it to the encoder, and downsampled the model’s output by the sampling rate of the original input. The \(i\)-th layer of encoder consists of a convolutional layer with a kernel size of \(K\), a stride of \(S\), and \(2^{i-1}H\) output channels, followed by a ReLU activation and a \(1 \times 1\) convolution with an output channel of \(2^iH\), and a GLU activation that converts the number of channels to \(2^{i-1}H\). A sequence model between the encoder and decoder is an LSTM network with two layers (each with \(2^{L-1}H\) hidden units). We adopted a causal version of DEMUCS; therefore, the LSTM was unidirectional. The \(i\)-th layer of decoder takes \(2^{L-i}H\) channels as input and performs \(1 \times 1\) convolution of \(2^{L-i+1}H\) channels, a GLU activation function for \(2^{L-i}H\) output channels, a transposed convolution of kernel size 8, stride 4, and \(2^{L-i-1}H\) output channels, and a ReLU function in sequence. There is no ReLU function in the last output layer. The experimental parameters are \(U = 4\), \(S = 4\), \(K = 8\), \(L = 5\), and \(H = 48\).

#### 4.3. Training Details

All our models were trained by the Adam optimizer with a step size of \(3 \times 10^{-4}\), a momentum of \(\beta_1 = 0.9\), and a denominator momentum \(\beta_2 = 0.999\). We used the Shift, Remix, and BandMask data augmentation methods proposed by Défossez et al. [20]. Shift is to apply a random shift from 0 to \(n\) seconds. Remix shuffles the noises in a batch to form new noisy mixtures. BandMask is a band-stop filter with a stop band between \(f_0\) and \(f_1\), sampled to remove 20% of frequencies in the mel scale. All audio is sampled at 16 kHz. When mixing two signals, we randomly chose an SNR between -5 and 5 dB.

The NyTT baseline was trained for 500 epochs using VoiceBank-DEMAND (noisy speech) and CHiME-3 (extraneous noise). This baseline NyTT model was also used as...
the initial teacher model. Each student model with the static teacher as TUP was trained for 500 epochs. Each student model with the exponentially moving average teacher as TUP was trained for 35 epochs. Because the training data did not contain clean speech, and there was no validation set for selecting the best model, we always tested the model after the last epoch in the experiments.

4.4. Results

We first compare the proposed models with the NyTT baseline. Two standardized metrics were used to evaluate the SE performance: perceptual evaluation of speech quality (PESQ) [24] and short-time objective intelligibility measure (STOI) [25]. The results are shown in Table 1. The top row shows the results of the NyTT baseline, also used as the initial teacher model for training our proposed models. The rows without * show the results of student models trained with a static teacher, while the rows with * show the results of student models trained with an exponentially moving average teacher. 1-stage refers to using only the student model for inference. 2-stage refers to using the initial teacher model for the first inference and the second inference.

Table 1. PESQ and STOI achieved by the baseline and our proposed models on Voice Bank+DEMAND. * indicates that the mean-updated strategy is used.

| Method       | PESQ  | STOI  |
|--------------|-------|-------|
|              | 1-stage | 2-stage | 1-stage | 2-stage |
| NyTT         | 2.198  | 2.208  | 0.932   | 0.932   |
| CTT-1        | 2.040  | 2.218  | 0.923   | 0.932   |
| CTT-2        | 2.068  | 2.222  | 0.928   | 0.932   |
| CTT-3        | 2.100  | 2.198  | 0.928   | 0.930   |
| NyTT-1       | 2.039  | 2.257  | 0.927   | 0.933   |
| NyTT-2       | 2.096  | 2.264  | 0.928   | 0.932   |
| NyTT-3       | 2.194  | 2.280  | 0.930   | 0.931   |
| CTT-1*       | 2.200  | 2.364  | 0.927   | 0.928   |
| CTT-2*       | 2.215  | 2.370  | 0.926   | 0.927   |
| CTT-3*       | 2.108  | 2.272  | 0.923   | 0.926   |
| NyTT-1*      | 2.215  | 2.371  | 0.927   | 0.928   |
| NyTT-2*      | 2.148  | 2.308  | 0.924   | 0.927   |
| NyTT-3*      | 2.108  | 2.273  | 0.923   | 0.926   |

From Table 1, we can see that in terms of PESQ, all student models trained with the static teacher performed worse than the baseline in single-stage inference. However, almost all student models outperformed the baseline in two-stage inference. Notably, the best performer among these models is NyTT-3, which was trained with a mixture of in-domain and extraneous noise as input. The mixture of in-domain and extraneous noise is relatively similar to the training noise of the baseline, thereby giving NyTT-3 comparable performance to the baseline. Compared with other models, NyTT-3 showed less improvement in two-stage inference. A similar trend was also observed from the results of CTT-3.

The student models trained with the exponentially moving average teacher outperformed the student models trained with the static teacher in terms of PESQ. Some models even outperformed the baseline in single-stage inference, and all models performed well in two-stage inference. However, in terms of STOI, the students models trained with the exponentially moving average teacher were worse than the student models trained with the static teacher and the baseline, which requires further study.

In Table 2, we compare our best model with previous supervised and unsupervised models. It can be seen that the supervised models are still more capable than the unsupervised models. When comparing the unsupervised models, it can be seen that both our self-implemented NyTT model and our best NyTT-1* model were inferior to the NyTT model in [13] in single-stage inference. Possible reasons are as follows. First, the model structure is slightly different. Second, we used a causal model architecture, which is less effective than a non-causal model architecture, but is more practical in real-world applications. Third, the original NyTT study used more training data than this work.

Table 2. PESQ achieved by various existing (un)supervised SE methods on Voice Bank+DEMAND. † indicates that external noisy data is used during training.

| Method                   | Unsupervised | 1-stage | 2-stage |
|--------------------------|--------------|---------|---------|
| NyTT† [13]               | ✓            | 2.160   | N/A     |
| MetricGAN-U (half) [17]  | ✓            | 2.450   | N/A     |
| MetricGAN-U (full) [17]  | ✓            | 2.130   | N/A     |
| NyTT (our implementation)| ✓            | 2.198   | 2.208   |
| Proposed (NyTT-1*)       | ✓            | 2.215   | 2.371   |

5. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed an unsupervised teacher-student training framework to mitigate the shortcomings of NyTT and other supervised methods. Our proposed method combines NyTT and RemixIT to generate extraneous noise. Our experiments show that the exponentially moving average teacher protocol is best suited for SE task. It is also found that two-stage inference helps our proposed framework to further improve the performance in terms of PESQ and STOI. In the future, we will conduct more experiments on various datasets with different unseen noise types. Furthermore, we will investigate why two-stage inference leads to a performance boost. It can provide another perspective for designing more efficient teacher-student frameworks without the need for multiple stages of inference.
6. REFERENCES

[1] Felix Weninger, Hakan Erdogan, Shinji Watanabe, Emmanuel Vincent, Jonathan Le Roux, John R. Hershey, and Björn Schuller, “Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR,” in Proc. LVA/ICA, 2015.

[2] Hassan Taherian, Zhong-Qiu Wang, Jorge Chang, and DeLiang Wang, “Robust speaker recognition based on single-channel and multi-channel speech enhancement,” IEEE Trans. Audio, Speech, Lang. Process., vol. 28, pp. 1293–1302, 2020.

[3] Deliang Wang and Jitong Chen, “Supervised speech separation based on deep learning: An overview,” IEEE Trans. Audio, Speech, Lang. Process., vol. 26, no. 10, pp. 1–27, 2018.

[4] Han Zhao, Shuaib Zarar, Ivan Tashev, and Chin-Hui Lee, “Convolutional-recurrent neural networks for speech enhancement,” in Proc. ICASSP, 2018.

[5] Hyeong-Seok Choi, Jang-Hyun Kim, Jaesung Huh, Adrian Kim, Jung-Woo Ha, and Kyoung Lee, “Phase-aware speech enhancement with deep complex-U-Net,” in Proc. ICLR, 2019.

[6] Yan Zhao, Zhong-Qiu Wang, and DeLiang Wang, “Two-stage deep learning for noisy-reverberant speech enhancement,” IEEE Trans. Audio, Speech, Lang. Process., vol. 27, no. 1, pp. 53–62, 2019.

[7] Haoyu Li, Szu-Wei Fu, Yu Tsao, and Junichi Yamagishi, “iMetricGAN: Intelligibility enhancement for speech-in-noise using generative adversarial network-based metric learning,” in Proc. Interspeech, 2020.

[8] Yuma Koizumi, Kohei Yatabe, Marc Delcroix, Yoshiki Masyuma, and Daiki Takeuchi, “Speech enhancement using self-adaptation and multi-head self-attention,” in Proc. ICASSP, 2020.

[9] Alexandre Défossez, Gabriel Synnaeve, and Yossi Adi, “Real-time speech enhancement in the waveform domain,” in Proc. Interspeech, 2020.

[10] Xiang Hao, Xiangdong Su, Radu Horaud, and Xiaofei Li, “FullSubNet: A full-band and sub-band fusion model for real-time single-channel speech enhancement,” in Proc. ICASSP, 2021.

[11] Ashutosh Pandey and DeLiang Wang, “Dense CNN with self-attention for time-domain speech enhancement,” IEEE Trans. Audio, Speech, Lang. Process., vol. 29, pp. 1270–1279, 2021.

[12] Guochen Yu, Yutian Wang, Chengshi Zheng, Hui Wang, and Qin Zhang, “CycleGAN-based non-parallel speech enhancement with an adaptive attention-in-attention mechanism,” in Proc. APSIPA ASC, 2021.

[13] Takuya Fujimura, Yuma Koizumi, Kohei Yatabe, and Ryoichi Miyazaki, “Noisy-target Training: A training strategy for DNN-based speech enhancement without clean speech,” in Proc. EUSIPCO, 2021.

[14] Xiaoyu Bie, Simon Leglaive, Xavier Alameda-Pineda, and Laurent Girin, “Unsupervised speech enhancement using dynamical variational autoencoders,” IEEE Trans. Audio, Speech, Lang. Process., vol. 30, pp. 2993–3007, 2022.

[15] Yang Xiang and Changchun Bao, “A parallel-data-free speech enhancement method using multi-objective learning cycle-consistent generative adversarial network,” IEEE Trans. Audio, Speech, Lang. Process., vol. 28, pp. 1826–1838, 2020.

[16] Viet Anh Trinh and Sebastian Braun, “Unsupervised speech enhancement with speech recognition embedding and disentanglement losses,” in Proc. ICASSP, 2022.

[17] Szu-Wei Fu, Cheng Yu, Kuo-Hsuan Hung, Mirco Ravaneli, and Yu Tsao, “MetricGAN-U: Unsupervised speech enhancement/de-reverberation based only on noisy/reverberated speech,” in Proc. ICASSP, 2022.

[18] Scott Wisdom, Efthymios Tzinis, Hakan Erdogan, Ron J. Weiss, Kevin Wilson, and John R. Hershey, “Unsupervised sound separation using mixture invariant training,” in Proc. NeurIPS, 2020.

[19] Efthymios Tzinis, Yossi Adi, Vamsi K. Ithapu, Buye Xu, Paris Smaragdis, and Anurag Kumar, “RemixIT: Continual self-training of speech enhancement models via bootstrapped remixing,” IEEE J. Sel. Top. Sig. Proc., vol. 14, no. 8, pp. 1–12, 2022.

[20] Alexandre Défossez, Nicolas Usunier, Léon Bottou, and Francis Bach, “Music source separation in the waveform domain,” arXiv preprint arXiv:1911.13254, 2019.

[21] Tomasz Grzywalski and Szymon Drgas, “Speech enhancement by multiple propagation through the same neural network,” MDPI Sens. J., vol. 22, no. 7, 2022.

[22] Cassia Valentini-Botinhao, Xin Wang, Shinji Takaki, and Junichi Yamagishi, “Investigating RNN-based speech enhancement methods for noise-robust text-to-speech,” in Proc. SSW, 2016.

[23] Jon Barker, Ricard Marxer, Emmanuel Vincent, and Shinji Watanabe, “The third CHiME speech separation and recognition challenge: Dataset, task and baselines,” in Proc. IEEE ASRU, 2015.

[24] Antony W. Rix, John G. Beerends, Michael P. Hollier, and Andries P. Hekstra, “Perceptual evaluation of speech quality (PESQ): A new method for speech quality assessment of telephone networks and codecs,” in Proc. ICASSP, 2001.

[25] Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen, “An algorithm for intelligibility prediction of time-frequency weighted noisy speech,” IEEE Trans. Audio, Speech, Lang. Process., vol. 19, no. 7, pp. 2125–2136, 2011.

[26] Santiago Pascual, Antonio Bonafonte, and Joan Serra, “SEGAN: speech enhancement generative adversarial network,” in Proc. Interspeech, 2017.

[27] Szu-Wei Fu, Chien-Feng Liao, Yu Tsao, and Shou-De Lin, “MetricGAN: Generative adversarial networks based black-box metric scores optimization for speech enhancement,” in Proc. ICML, 2019.

[28] Ruizhe Cao, Sherif Abdalatib, and Bin Yang, “CMGAN: Conformer-based MetricGAN for speech enhancement,” in Proc. Interspeech, 2022.