INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Fine-tuning pre-trained voice conversion model for adding new target speakers with limited data

Takeshi Koshizuka, Hidefumi Ohmura, Kouichi Katsurada

Department of Information Sciences, Tokyo University of Science, Japan
takeshikoshizuka938444@gmail.com, hidefumi.ohmura@gmail.com, katsurada@rs.tus.ac.jp

Abstract

Voice conversion (VC) is a technique that converts speaker-dependent non-linguistic information into that of another speaker, while retaining the linguistic information of the input speech. A typical VC system comprises two modules: an encoder module that removes speaker individuality from the input speech and a decoder module that incorporates another speaker's individuality in synthesized speech. This paper proposes a training method for a vocoder-free any-to-many encoder-decoder VC model with limited data. Various pre-training techniques have been proposed to solve problems training to limited training data; some of these techniques employ the text-to-speech (TTS) task for pre-training. We pre-train the decoder module in the voice conversion task for growing our pre-training technique into continuously adding target speakers to the VC system. The experimental results show that good conversion performance can be achieved by conducting VC-based pre-training. We also confirmed that the rehearsal and pseudo-rehearsal methods can effectively fine-tune the model without degrading the conversion performance of the pre-trained target speakers.

Index Terms: Any-to-Many Voice Conversion, Pre-training, Addition of target speakers with limited data

1. Introduction

Voice conversion (VC) is a technique that converts speaker-dependent non-linguistic information into that of another speaker, while retaining the linguistic information of the input speech. A successful VC approach involves statistical models based on the gaussian mixture model (GMM) [1, 2, 3] and neural network (NN)-based models [4, 5, 6]. Owing to the developments in deep generative models, recent studies have examined the variational autoencoder (VAE)-based model [7, 8] and generative adversarial network (GAN)-based model [9, 10]. In general, VC using an encoder-decoder model, such as the VAE, involves two modules: an encoder module that removes speaker individuality from the speech and a decoder module that incorporates another speaker's individuality in the synthesized speech. Conventional statistical VC models typically convert speech waveforms into feature parameters and then employ a vocoder to generate speech. Recent studies have proposed vocoder-free VC, which integrates the vocoder and conversion models to afford more flexible conversion. For real-world applications, large-scale VC models typically require significant amounts of training data. However, it is difficult to collect the required speech data from specific speakers.

Pre-training is one of the solutions for this problem. Using pre-training techniques [11, 12], VC models can be trained with limited speech data, as they can preliminarily learn the model parameters required for a variety of speech processing tasks. For instance, [12] uses the vq-wav2vec model [13] and the transformer-seq2seq model [14] pre-trained with the automatic speech recognition (ASR) corpus and the TTS corpus, respectively. [12] proposed a pre-training technique that makes the model more robust against limited training data. This paper proposes a novel training method for a vocoder-free, any-to-many VC model with limited data. We pre-train the decoder module with the VC task and subsequently fine-tune the decoder module for new target speakers with limited data. Pre-training the decoder module with the VC task is expected to be the first step in developing a new training method for VC models with a large number of target speakers by sequentially adding target speakers to the model.

During pre-training with the VC task, the fine-tuning phase can be considered as training for adding new target speakers to the pre-trained VC model. In this context, it is desirable to fine-tune the model for new target speakers, without deteriorating the conversion performance of the pre-trained target speakers. A well-known technique for learning a new task without deteriorating the performance of the previous task is continual learning (life-long learning). This paper proposes a rehearsal method and a pseudo-rehearsal method, both inspired by the approach of continual learning. The rehearsal method explicitly retrain on a subset of speech data from pre-trained speakers, while performing fine-tuning for new target speakers. Although the rehearsal method is simple and straightforward, it cannot be applied when the speech data used in pre-training are unavailable. To address this, we introduce the pseudo-rehearsal method. The pseudo-rehearsal method uses the pre-trained VC model to generate the pseudo-data of the pre-trained speakers. Lastly, we fine-tune the model using both the pseudo-data and the data from the new target speakers. The proposed pre-training method with the VC task can significantly improve VC performance, even when the training data from the new target speakers are limited. We also show that the rehearsal and pseudo-rehearsal methods can fine-tune the model for new target speakers, while reducing the degradation in conversion performance of the pre-trained target speakers.

2. Related work

2.1. VC model using vq-wav2vec

The VC model proposed by Huang et al. [12] consists of the pre-trained vq-wav2vec [13] and transformer-TTS based seq2seq (sequence-to-sequence) [14] models. The seq2seq model employs a series of acoustic units of phoneme level duration from vq-wav2vec as an input and outputs acoustic features of the target speaker's speech. [12] pre-trained the seq2seq model in two steps. In the first step, the decoder is pre-trained using a large-scale text-to-speech (TTS)-corpus to train a conventional TTS model. In the second step, the encoder is trained using reconstruction loss, while the pre-trained decoder remains unchanged. The vq-wav2vec model used by Huang et al. learns

http://dx.doi.org/10.21437/Interspeech.2021-244

representations of the linguistic information in speech by solving a self-supervised context prediction task [15]. The network comprises an encoder module for feature extraction, a VQ module to discretizing representations, and an aggregator module to output context representations. In accordance with [13], mode collapse is mitigated by partitioning each frame's representations into multiple groups. Subsequently, the VQ module independently quantizes the partitioned representations, and we obtain a series of acoustic units for each group. Huang et al. proposed a method to use the embedding representation of an acoustic unit separately for each group. The experiments show that the pre-training technique makes the model more robust against limited training data.

2.2. Continual learning

In certain cases, it is necessary to train a VC model for new target speakers by using the knowledge obtained during the training for previous target speakers. However, training the model for new target speakers can cause catastrophic forgetting [16, 17], which is a degradation of performance for a previously trained target speaker. Continual learning is a concept for sequentially training a model for multiple tasks, while avoiding catastrophic forgetting. Continual learning continually accumulates knowledge over various tasks, without retraining the model from scratch. One approach to continual learning is the rehearsal technique [18, 19, 20], which stores some samples of the training data form previous tasks and uses them during the training for the new tasks. The rehearsal technique is a suitable solution to circumvent catastrophic forgetting; however, it cannot be used when the training data for previous tasks are unavailable. In this case, the pseudo-rehearsal technique[21, 22], which employs pseudo-training samples of previous tasks, is adopted. Deep generative replay [22] is a framework categorized as a pseudo-rehearsal technique. Deep generative replay features a cooperative, dual model architecture consisting of a deep generative model (generator) and a task solving model (solver). This generator-solver pair can produce pairs of pseudo-data and the desired output for previous tasks as required and can also retrain the model on these produced pairs, while updating the generator and solver.

3. System description

3.1. Model description

Our VC model consists of a vq-wav2vec [13] encoder, which is used in the VC model described in section 2.1 [12] and an autoregressive decoder module with a structure similar to the RNN_MS [23] used in [24]. The decoder module is composed of a conditioning sub-network and an autoregressive model. Figure 1 illustrates our VC model. Gated recurrent unit (GRU) blocks are added to the conditioning sub-network of the decoder module. The GRU blocks, which comprise two-layer bidirectional GRUs, employ the combined representations of code embedding and the outputs from the previous GRUs as the inputs. We aim to represent more complex speech features by using M layers of GRU blocks. We determined that $M = 2$ was optimal for good VC performance. The speaker embedding layer and the code embedding layers refer to a fixed size dense vector in their respective lookup tables.

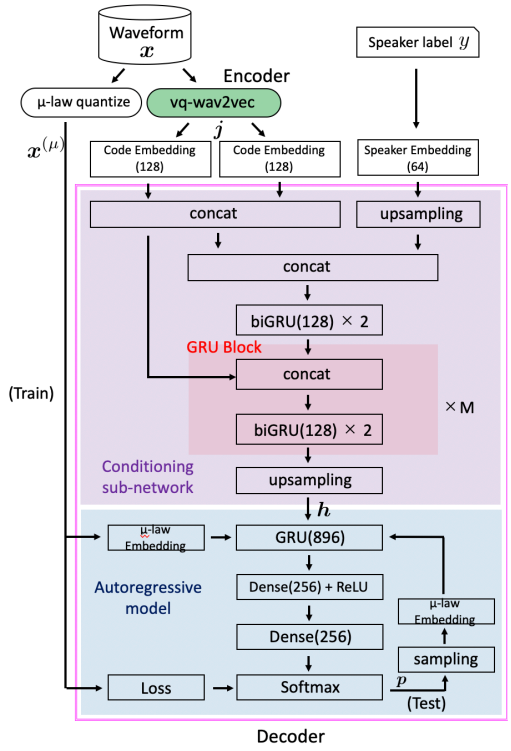

Figure 1: *Architecture of voice conversion model*

3.2. Training details

We train the model to minimize the negative log-likelihood ℓ:

$$\ell(\boldsymbol{x}, y) := \frac{1}{T}\sum_{t=2}^{T+1} -\log p(x_t^{(\mu)} \mid x_{t-1}^{(\mu)}, \boldsymbol{j}, y) \qquad (1)$$

where $\boldsymbol{x} = (x_1, \ldots, x_T)$ is a sample of an audio waveform sequence, $\boldsymbol{x}^{(\mu)} = (x_1^{(\mu)}, \ldots, x_T^{(\mu)})$ is a μ-law quantized audio waveform sequence, $\boldsymbol{j} = \mathrm{Enc}(\boldsymbol{x})$ is a series of acoustic units encoded from $\boldsymbol{x}$ by the encoder module (vq-wav2vec), and y is the speaker label corresponding to $\boldsymbol{x}$.

3.3. Testing details

In the testing phase, the vq-wav2vec model encodes a source audio waveform into the embedding series of an acoustic unit. Next, the conditioning sub-network employs the embedding series of the acoustic unit and speaker embedding of a target speaker as the inputs, and it outputs the conditioning input features for audio generation. Thereafter, the autoregressive model employs the conditioning features $\boldsymbol{h}$ obtained from the conditioning sub-network as the inputs and outputs the probability parameters $\boldsymbol{p}$ for the distribution of the μ-law quantized audio samples. Finally, converted audio samples are generated via sampling from the distribution of the μ-law quantized audio samples and by decoding μ-law quantization.

4. Proposed training with limited data

4.1. Pre-training with the VC task and fine-tuning

We propose a novel training method that pre-trains the encoder and decoder modules separately. First, we pre-train the encoder module on large-scale data by performing a pretext task, such

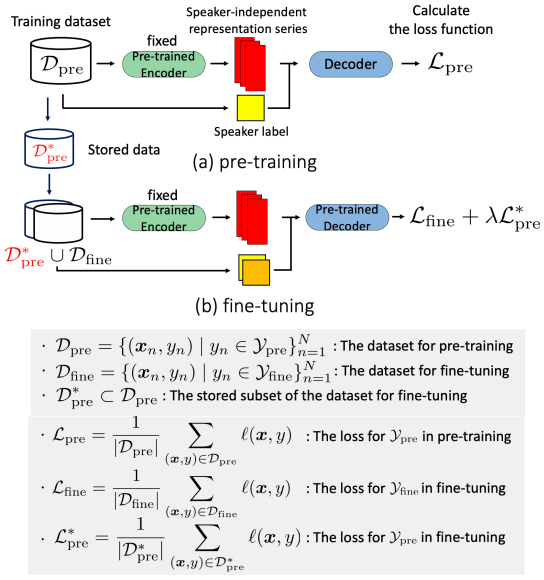

Figure 2: *Algorithm of rehearsal method*

as self-supervised context prediction. Next, we pre-train the decoder module with the VC task for speakers using large amounts of speech data as target speakers while the encoder module remains unchanged. Finally, we fine-tune the decoder module for the new target speakers with limited data.

4.2. Fine-tuning for adding new target speakers

Considering the fine-tuning as training for adding new target speakers to the pre-trained VC model, it is desirable to fine-tune the model for new target speakers without degrading the conversion performance for pre-trained target speakers. However, the proposed training method, described in section 4.1, typically results in a degradation in the performance for the pre-trained target speakers. To prevent this degradation, we introduce the rehearsal method and the pseudo-rehearsal method, both inspired by the approach for continual learning.

Figure 2 shows the algorithm of the rehearsal method and the mathematical notations of datasets and loss terms. The rehearsal method stores a subset $\mathcal{D}^*_{\text{pre}}$ of the dataset used for pre-training $\mathcal{D}_{\text{pre}}$ and explicitly retrain the decoder module for the pre-trained target speakers by minimizing $\mathcal{L}^*_{\text{pre}}$, while performing fine-tuning for the new target speakers. It should be noted that $\mathcal{Y}_{\text{pre}}$ and $\mathcal{Y}_{\text{fine}}$ are the speaker label sets for pre-training and fine-tuning, respectively. The rehearsal method is a simple and straightforward approach for preventing degradation in the performance for the pre-trained target speakers. However, it cannot be used when the dataset for pre-training $\mathcal{D}_{\text{pre}}$ is unavailable due to some real-world limitations, including the privacy issues. We employ the pseudo-rehearsal method to overcome this problem.

Figure 3 shows the algorithm of the pseudo-rehearsal method and the mathematical notations of datasets and loss terms. The pseudo-rehearsal method generates pseudo-speech data of the pre-trained speakers $\tilde{\boldsymbol{x}}$ from any speaker's speech data $\boldsymbol{x} \in \mathcal{X}_g$ by using the pre-trained VC model. Speaker-independent representations $\boldsymbol{j}$ encoded by the encoder module are also stored. Subsequently, we fine-tune the decoder module using both the stored pseudo-speech dataset $\mathcal{S}_{\text{pre}}$ and small dataset of the new target speakers, i.e., $\mathcal{D}_{\text{fine}}$.

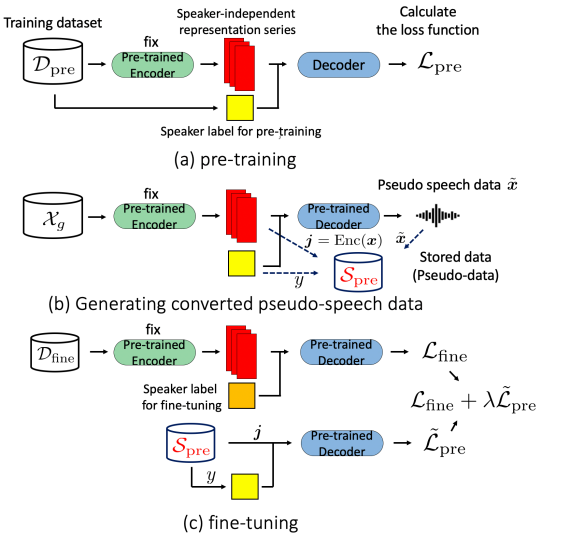

Figure 3: *Algorithm of pseudo-rehearsal method*

5. Experiments setting

5.1. Evaluation metrics

We carried out three objective evaluations between the converted speech and the ground truth: the mel-cepstrum distortion (MCD), the word error rate (WER), and the character error rate (CER). For calculating the WER and CER, we used a transformer-based end-to-end ASR engine trained on the open-source ESPnet toolkit [25, 26]. We also conducted a subjective evaluation based on naturalness and speaker similarity among the samples. For naturalness, we conducted mean opinion score (MOS) test using a five-point-scale, ranging from 1 (completely unnatural) to 5 (completely natural). Furthermore, for speaker similarity, subjects were asked to listen to the audio pairs and judge whether they belonged to the same speaker; a four-point scale, ranging from 1 (different: absolutely sure) to 4 (same: absolutely sure), was used for this test.

5.2. Evaluation setting

We evaluated the proposed training methods, described in section 4, using the CMU ARCTIC database [27], which contains parallel recordings of professional US English speakers sampled at 16 kHz. In the experiments, we used the VC model comprising the pre-trained vq-wav2vec and the RNN_MS-like decoder modules with a two-layer GRU block. For the vq-wav2vec model, we used the publicly available pre-trained model [1] which employs the complete 960-h Librispeech dataset [28] for training. We compared the evaluation results of the three training methods, i.e., (*Fine-tuning*, *Rehearsal*, and *Pseudo-rehearsal*), and compared them with three baselines

[1] https://github.com/pytorch/fairseq/tree/master/examples/wav2vec

(*Baseline (50)*, *Baseline (932)*, and *Baseline (493)*) and the ground truth.

Fine-tuning pre-trains the decoder module with the VC task and fine-tunes the entire decoder module for $\mathcal{Y}_{\text{fine}}$, as described in section 4.1. To confirm the performance of the fine-tuning method in adding new target speakers, we evaluated the rehearsal method and the pseudo-rehearsal method, denoted by *Rehearsal* and *Pseudo-rehearsal*, respectively. The training dataset for pre-training $\mathcal{D}_{\text{pre}}$ consists of 593 utterances (493 for training, and 100 for validation) per target speaker in $\mathcal{Y}_{\text{pre}}$: two male speakers (aew and awb) and two female speakers (ljm and lnh). The training dataset for fine-tuning $\mathcal{D}_{\text{fine}}$ consists of 70 utterances (50 for training, and 20 for validation) per target speaker in $\mathcal{Y}_{\text{fine}}$: a male speaker (rms) and a female speaker (slt). The unlabeled speech dataset for pseudo-speech data generation $\mathcal{X}_g$ consists of 280 utterances of a male speaker (gka). Baseline (932) and Baseline (493) are models trained with 932 utterances per target speaker in $\mathcal{Y}_{\text{fine}}$ and 493 utterances per target speaker in $\mathcal{Y}_{\text{pre}}$ without pre-training, respectively. These models were trained using the maximum number of training utterances available for each target speaker in the dataset, and therefore their evaluation results constitute the upper bounds of performance. By contrast, Baseline (50) is a model trained with the small dataset for fine-tuning $\mathcal{D}_{\text{fine}}$ without pre-training. This is a baseline for confirming the degradation in performance when only the limited data are available, and its evaluation results constitute the lower bounds of performance. During the testing phase, we used a male speaker (bdl) and a female speaker (clb) as the unseen source speakers involved in the conversion. For a objective evaluation of each VC system, we evaluated 100 utterances per source-target speaker pair. For a subjective evaluation, we evaluated 16 random utterances per VC system. We used Amazon Mechanical Turk (Mturk) and recruited fifty listeners to conduct the subjective evaluations.

We trained the decoder module with a batch size of 32, and the input speech had a fixed length of 5,120 samples. We set the number of initial iterations to 160,000 and used the early stopping method, which monitors the validation data loss and stops training at the lowest point of loss. Moreover, we used the Adam optimizer and set the initial learning rate to 4.0×10^{-4} and the decay rate to $\beta_1 = 0.9$, $\beta_2 = 0.999$. The learning rate is scheduled to be halved when the number of iterations reached 50,000, 75,000, 100,000, and 125,000. For the rehearsal and pseudo-rehearsal methods, the ratio parameter $\lambda = \frac{|\mathcal{Y}_{\text{fine}}|}{|\mathcal{Y}_{\text{pre}}|} = \frac{1}{2}$ was used in fine-tuning.

6. Evaluation results

6.1. Effectiveness of pre-training with the VC task

First, we compared the performance *Fine-tuning* and *Baseline (50)* for the target speakers $\mathcal{Y}_{\text{fine}}$. As shown in Table 1, *Fine-tuning* significantly outperforms *Baseline (50)*, especially in terms of the WER, CER, and naturalness. These results indicate that pre-training with the VC task enables the decoder module to learn VC for the new target speakers $\mathcal{Y}_{\text{fine}}$ with good performance by only using limited data.

6.2. Evaluation of *Rehearsal* and *Pseudo-rehearsal*

Next, we compared the performance of *Rehearsal*, *Pseudo-rehearsal*, and *Fine-tuning*, and the results are shown in Table 1. With regard to the target speakers $\mathcal{Y}_{\text{pre}}$, *Rehearsal* demonstrates substantial improvement in all measures, as compared to *Fine-tuning*. Furthermore, for the target speakers $\mathcal{Y}_{\text{fine}}$, *Rehearsal* exhibits values similar to those of *Fine-tuning* for all measures. These results indicate that the rehearsal method can train additional target speakers by using the limited data alone, without deteriorating the performance for the pre-trained target speakers $\mathcal{Y}_{\text{pre}}$.

Finally, for the target speakers $\mathcal{Y}_{\text{pre}}$, *Pseudo-rehearsal* achieves better values for MCD and similarity than *Fine-tuning*, but worse values for WER, CER, and naturalness. The primary reason for this deterioration is likely the inclusion of poor-quality converted speech in the stored pseudo-speech data $\mathcal{S}_{\text{pre}}$. With regard to the target speakers $\mathcal{Y}_{\text{fine}}$, *Pseudo-rehearsal* achieves better values for all measures, as compared to *Fine-tuning*. Although this improvement was unexpected when introducing the pseudo-rehearsal method, but it can be attributed to the effect of data augmentation. These results indicate that the pseudo-rehearsal method can train additional target speakers with good performance by only using the limited data, while preventing degradation in the conversion performance of speaker individuality for the target speakers $\mathcal{Y}_{\text{pre}}$.

Table 1: *Evaluation results of the proposed training method*

(a) *Objective evaluation results of the proposed training method.*

Training methods	$\mathcal{Y}_{\text{fine}}$(rms, slt)			$\mathcal{Y}_{\text{pre}}$(aew, lnh, awb, ljm)		
	MCD	WER	CER	MCD	WER	CER
Fine-tuning	8.00	12.3	8.48	8.67	15.0	11.5
Rehearsal	8.07	13.1	8.98	**7.68**	**11.9**	**9.36**
Pseudo-rehearsal	**7.88**	**8.23**	**5.78**	8.02	17.8	13.6
Baseline (50)	8.05	21.6	15.4	-	-	-
Baseline (932)	7.72	9.55	5.95	-	-	-
Baseline (493)	-	-	-	7.70	12.4	9.14
Ground truth	-	4.23	2.35	-	-	-

(b) *Subjective evaluation results of the proposed training method with 95% confidence intervals.*

Training methods	$\mathcal{Y}_{\text{fine}}$(rms, slt)		$\mathcal{Y}_{\text{pre}}$(aew, lnh, awb, ljm)	
	Naturalness	Similarity	Naturalness	Similarity
Fine-tuning	3.55 ± 0.07	2.70 ± 0.08	3.06 ± 0.08	1.76 ± 0.07
Rehearsal	3.39 ± 0.07	2.76 ± 0.08	$\mathbf{3.44 \pm 0.07}$	$\mathbf{2.87 \pm 0.07}$
Pseudo-rehearsal	$\mathbf{3.73 \pm 0.06}$	$\mathbf{2.92 \pm 0.07}$	2.98 ± 0.08	2.34 ± 0.08
Baseline (50)	1.74 ± 0.07	2.13 ± 0.06	-	-
Baseline (932)	3.90 ± 0.06	2.85 ± 0.08	-	-
Baseline (493)	-	-	3.46 ± 0.07	2.62 ± 0.08
Ground truth	4.52 ± 0.05	3.52 ± 0.06	4.25 ± 0.06	3.43 ± 0.06

7. Conclusion

This paper proposed a training method that pre-trains the decoder module with the VC task to train the encoder-decoder VC model using limited data. We also introduced the rehearsal and pseudo-rehearsal methods for adding new target speakers to the model by using limited data, without deteriorating the performance for the pre-trained target speakers. The evaluation results showed that the proposed pre-training method with the VC task could significantly improve performance, even when training with limited training data, and that the rehearsal method and the pseudo-rehearsal method can fine-tune the model for new target speakers, while preventing a degradation in the performance for pre-trained target speakers. In the future, we plan to develop a training method for adding target speakers to the trained VC model with limited data at lower computational and memory costs.

8. References

[1] Y. Stylianou, O. Cappé, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Transactions on speech and audio processing*, vol. 6, no. 2, pp. 131–142, 1998.

[2] A. Kain and M. W. Macon, "Spectral voice conversion for text-to-speech synthesis," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, vol. 1. IEEE, 1998, pp. 285–288.

[3] T. Toda, A. W. Black, and K. Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[4] S. Desai, E. V. Raghavendra, B. Yegnanarayana, A. W. Black, and K. Prahallad, "Voice conversion using artificial neural networks," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 3893–3896.

[5] L.-H. Chen, Z.-H. Ling, L.-J. Liu, and L.-R. Dai, "Voice conversion using deep neural networks with layer-wise generative training," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1859–1872, 2014.

[6] F.-L. Xie, Y. Qian, Y. Fan, F. K. Soong, and H. Li, "Sequence error (se) minimization training of neural network for voice conversion," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[7] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from non-parallel corpora using variational auto-encoder," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2016, pp. 1–6.

[8] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "Acvae-vc: Non-parallel voice conversion with auxiliary classifier variational autoencoder," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 9, pp. 1432–1443, 2019.

[9] Y. Saito, S. Takamichi, and H. Saruwatari, "Statistical parametric speech synthesis incorporating generative adversarial networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 1, pp. 84–96, 2017.

[10] T. Kaneko and H. Kameoka, "Cyclegan-vc: Non-parallel voice conversion using cycle-consistent adversarial networks," in *2018 26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018, pp. 2100–2104.

[11] W.-C. Huang, T. Hayashi, Y.-C. Wu, H. Kameoka, and T. Toda, "Voice Transformer Network: Sequence-to-Sequence Voice Conversion Using Transformer with Text-to-Speech Pretraining," in *Proc. Interspeech 2020*, 2020, pp. 4676–4680.

[12] W.-C. Huang, Y.-C. Wu, T. Hayashi, and T. Toda, "Any-to-one sequence-to-sequence voice conversion using self-supervised discrete speech representations," *arXiv preprint arXiv:2010.12231*, 2020.

[13] A. Baevski, S. Schneider, and M. Auli, "vq-wav2vec: Self-supervised learning of discrete speech representations," in *International Conference on Learning Representations*, 2020.

[14] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, "Neural speech synthesis with transformer network," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 6706–6713.

[15] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[16] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *Psychology of learning and motivation*. Elsevier, 1989, vol. 24, pp. 109–165.

[17] R. Ratcliff, "Connectionist models of recognition memory: constraints imposed by learning and forgetting functions." *Psychological review*, vol. 97, no. 2, p. 285, 1990.

[18] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, "icarl: Incremental classifier and representation learning," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010.

[19] D. Isele and A. Cosgun, "Selective experience replay for lifelong learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[20] D. Rolnick, A. Ahuja, J. Schwarz, T. Lillicrap, and G. Wayne, "Experience replay for continual learning," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., vol. 32. Curran Associates, Inc., 2019.

[21] A. Robins, "Catastrophic forgetting, rehearsal and pseudorehearsal," *Connection Science*, vol. 7, no. 2, pp. 123–146, 1995.

[22] H. Shin, J. K. Lee, J. Kim, and J. Kim, "Continual learning with deep generative replay," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 2994–3003.

[23] J. Lorenzo-Trueba, T. Drugman, J. Latorre, T. Merritt, B. Putrycz, R. Barra-Chicote, A. Moinet, and V. Aggarwal, "Towards Achieving Robust Universal Neural Vocoding," in *Proc. Interspeech 2019*, 2019, pp. 181–185.

[24] B. van Niekerk, L. Nortje, and H. Kamper, "Vector-quantized neural networks for acoustic unit discovery in the zerospeech 2020 challenge," *Proc. Interspeech 2020*, pp. 4836–4840, 2020.

[25] T. Hayashi, R. Yamamoto, K. Inoue, T. Yoshimura, S. Watanabe, T. Toda, K. Takeda, Y. Zhang, and X. Tan, "Espnet-tts: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7654–7658.

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

[27] J. Kominek, A. W. Black, and V. Ver, "Cmu arctic databases for speech synthesis," 2003.

[28] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015.