Interspeech 2022
18-22 September 2022, Incheon, Korea

Self supervised learning for robust voice cloning

Konstantinos Klapsas[1], *Nikolaos Ellinas*[1], *Karolos Nikitaras*[1], *Georgios Vamvoukakis*[1], *Panos Kakoulidis*[1], *Konstantinos Markopoulos*[1], *Spyros Raptis*[1], *June Sig Sung*[2], *Gunu Jho*[2], *Aimilios Chalamandaris*[1], *Pirros Tsiakoulis*[1]

[1]Innoetics, Samsung Electronics, Greece
[2]Mobile Communications Business, Samsung Electronics, Republic of Korea

{n.ellinas, g.vamvouk, p.kakoulidis, k.markop, s.raptis, js6.sung, gunu.jho, aimilios.ch, p.tsiakoulis} @samsung.com,
{k.klapsas, k.nikitaras} @partner.samsung.com

Abstract

Voice cloning is a difficult task which requires robust and informative features incorporated in a high quality TTS system in order to effectively copy an unseen speaker's voice. In our work, we utilize features learned in a self-supervised framework via the Bootstrap Your Own Latent (BYOL) method, which is shown to produce high quality speech representations when specific audio augmentations are applied to the vanilla algorithm. We further extend the augmentations in the training procedure to aid the resulting features to capture the speaker identity and to make them robust to noise and acoustic conditions. The learned features are used as pre-trained utterance-level embeddings and as inputs to a Non-Attentive Tacotron based architecture, aiming to achieve multispeaker speech synthesis without utilizing additional speaker features. This method enables us to train our model in an unlabeled multispeaker dataset as well as use unseen speaker embeddings to copy a speaker's voice. Subjective and objective evaluations are used to validate the proposed model, as well as the robustness to the acoustic conditions of the target utterance.

Index Terms: voice cloning, self supervised, byol-a

1. Introduction

Speech synthesis research has advanced from high quality attentive acoustic models, like Tacotron [1, 2], to state-of-the-art models that do not depend on attention, like Non-attentive Tacotron [3] and FastSpeech [4]. By providing the acoustic model with learnable speaker embeddings it is shown that multispeaker speech synthesis is feasible with high fidelity [5, 6] and can be extended with few modifications to multilingual datasets [7].

The task of cloning the voice of speakers not included in the training set, requires representations that can generalize to unseen data. Especially interesting are cases where the unseen speaker's utterances are limited [8]. Other parameters such as speaker style cloning can also be factored in this task [9].

1.1. Related Work

Since neural Text-to-Speech (TTS) models are capable of producing speech conditional to several factors, a speaker embedding is an adequate representation in order to differentiate between various speakers in a single model. There are many ways of adapting a multispeaker model to a new speaker, for example fine-tuning [10, 11] is a standard approach that uses the target speaker's data to continue training of the base model. In [12] a multi-stage speaker adaptation method is also proposed, whereas in [13] meta-learning is used in order to increase the generalization capability of the model. Adaptation is also shown to work effectively in multilingual setups [14, 15].

Alternatively, a speaker encoder that can directly predict a speaker embedding from audio can be applied with success to limited data scenarios [8]. There is a lot of research on generalized speaker representations applicable to many other tasks, such as x-vectors for speaker recognition [16] and d-vectors for speaker verification [17]. These can be successfully applied in the cloning task [18] by using them as a pre-trained model in a transfer learning approach. Instead of a fixed speaker embedding, in [19] a variable-length embedding is extracted from audio on-the-fly and conditions the decoder on speaker-dependent features producing high quality results. The speaker identity can also be sensitive to acoustic conditions [20], so domain adversarial training can be employed in speaker adaptation and speaker encoding [21] in order to perform voice cloning from noisy samples.

Self-supervised learning is a machine learning approach where the model trains itself by leveraging part of the data to generate supervisory signals for the task at hand. Pre-training methods for speech are used to obtain representations that are useful for a variety of possible tasks [22]. State-of-the-art models like wav2vec 2.0 [23] follow a contrastive approach, while we focus on the non-contrastive methods which do not include negative samples. Specifically, BYOL [24] and SimSiam [25] pair the representations learned by two neural networks by applying augmentations on the training data. BYOL has been adapted for audio data [26] in order to generate general purpose embeddings, which are shown to be useful in a large variety of tasks. We aim at applying and extending this method to the task of voice cloning and we show that it allows learning of meaningful speaker representations that can directly condition a high quality TTS model.

1.2. Our Contributions

To the best of our knowledge, this is the first application of learned self-supervised features on neural voice cloning. The main contributions of the paper are as follows:

- We present a voice cloning algorithm that can be trained on an unlabeled dataset with an arbitrary number of speakers.
- We demonstrate that our model is able to perform voice cloning with similar performance to a d-vectors baseline while using only a fraction of the training dataset.
- We incorporate additional augmentations in order to

Copyright © 2022 ISCA

10.21437/Interspeech.2022-10856

make the learned self-supervised representations robust to acoustic conditions and prosodic variations, enabling robust voice cloning.

2. Method

2.1. Overview

Our approach is based on a Non-attentive Tacotron TTS architecture [3], adapted for producing features for the LPCNet vocoder [27] and conditioned on pre-trained self-supervised features. The pre-training method we use in our work is an adaptation of the Bootstrap Your Own Latent (BYOL) algorithm [24], an effective self-supervised learning method that produces meaningful representations. In previous work, the original algorithm was adapted for learning audio representations by introducing audio-related augmentations and was called BYOL for Audio (BYOL-A) [26]. Given that this algorithm has been shown to be effective in the task of speaker identification without a labeled dataset, we simply condition our TTS system on the audio representations as speaker embeddings to achieve voice cloning.

In the following paragraphs, the training algorithm is explained in brief, along with the additional augmentations we included to better help the model capture speaker identity, and make it more robust to noise in the reference samples.

2.2. BYOL for audio

BYOL training consists of the simultaneous training of two neural networks, the online and target networks. The two networks have the same architecture but use a different set of weights, denoted as θ for the online and as ξ for the target network. Specifically, both networks consist of a representation encoder f, and a projector g, namely f_θ, g_θ for the online and f_ξ, g_ξ for the target network. Additionally, the online network has an additional prediction module q_ξ.

The training is done by first producing two augmented views of the audio x, $u \triangleq t(x)$ and $u' \triangleq t'(x)$ where t and t' are sampled from the distribution of audio augmentations $t, t' \sim T$. The online network is then used to output the representation $y_\theta \triangleq f_\theta(u)$, and a projection $z_\theta \triangleq g_\theta(y)$. It is important to note that during inference, only the representation $y_\theta = f_\theta(u)$ is used.

Similarly, the target network is used to output the target projection $z'_\xi \triangleq g_\xi(f_\xi(u'))$ from the second augmented view. The additional prediction module is used on the online prediction to get the prediction $q_\theta(z_\theta)$ of z'_ξ, which is ℓ_2 normalized along with the target projection, $\bar{q}_\theta(z_\theta) \triangleq q_\theta(z_\theta)/\|q_\theta(z_\theta)\|_2$ and $\bar{z}'_\xi \triangleq z'_\xi/\|z'_\xi\|_2$.

The loss is defined as:

$$\mathcal{L}_{\theta,\xi} \triangleq \|\bar{q}_\theta(z_\theta) - \bar{z}'_\xi\|_2 \tag{1}$$

In order for the loss to be symmetric with respect to the augmentations, the u' augmentation is also fed to the online network and u to the target network and the loss is recomputed to get $\mathcal{L}^{\sim}_{\theta,\xi}$. The final loss is $\mathcal{L}^{BYOL}_{\theta,\xi} = \mathcal{L}^{\sim}_{\theta,\xi} + \mathcal{L}_{\theta,\xi}$.

Only the online network is updated in order to minimize this training loss. The parameters of the target network are updated as an exponential moving average of the online parameters [28], as follows:

$$\xi \leftarrow \tau\xi + (1-\tau)\theta \tag{2}$$

where $\tau \in [0, 1]$ is the target decay rate, which is set to 0.99 in our experiments. It has been shown [24, 29] that this training procedure is sufficient to avoid collapsed solutions such as constant representations, since the updates to the target network parameters ξ are not in general in the direction of $\nabla_\xi \mathcal{L}^{BYOL}_{\theta,\xi}$, due to the stop grad operation in the target network.

The input to the networks in the case of audio data is a one second segment of log-mel spectrograms.

2.3. BYOL-A Augmentations

2.3.1. Pre and post Normalization

Both pre and post-augmentation normalization is done on the samples, $\tilde{x} = \frac{x-\mu}{\sigma}$ where μ is the mean and σ is the standard variation. Pre-normalization is done using the statistics of the whole dataset, while post-normalization is done using the statistics of the current batch.

2.3.2. Mixup

A Mixup block [30] is utilized, which mixes randomly selected past inputs with the current input in a small ratio. The past inputs are therefore used as a background sound, something which helps the network to learn representations only of the foreground acoustic event.

Since the acoustic features are log scaled, it is first converted to a linear scale, before the mixup is applied and then converted back to the log domain. The mixing ratio that is used is sampled from a Uniform distribution $U(0, \alpha)$ where α is a hyper-parameter that is 0.4 in our experiments.

While in the original implementation of BYOL-A the main purpose of the mixup block is to discriminate between foreground and background, we find that it is also crucial to our approach by encouraging the model to learn more discriminative features between speakers. This stems from the fact that the two paths of the BYOL-A training may contain past inputs sampled from different speakers.

2.3.3. RRC

Random Resize Crop is an image augmentation technique that is adapted for audio by applying it to the Mel Spectrogram of an audio segment. It can be conceived as an approximation of pitch shifting and time stretching.

The procedure consists of sampling a random crop from the log mel spectrogram. Given a number of frequency bins F and of time frames T, the size of the crop area is randomly sampled as:

$$\begin{aligned} F_C &= \lfloor \min\left(U(h_1, h_2), 1.0\right) \times F \rfloor \\ T_C &= \lfloor U(w_1, w_2) \times T \rfloor \end{aligned} \tag{3}$$

where F_c and T_C are the number of frequency and time frames respectively and $[h_1, h_2]$ and $[w_1, w_2]$ are the ranges of scaling that are being used for frequency and time. The default values that are used are 0.6 and 1.5 for both dimensions, which means that the new crop area may be outside of the boundaries of the original spectrogram. In this case zeros are used to fill the required area.

2.3.4. Gaussian Noise

A final augmentation is a Gaussian augmentation block that interpolates between training data and noise sampled from a normal distribution. The purpose of this augmentation is similar to the mixup augmentation. The Gaussian noise is sampled from

$N(0, 0.04)$ and added to the log-domain with the same log-exp trick as the mixup augmentation.

2.4. Additional Augmentations for Robust Voice Cloning

2.4.1. Prosodic Augmentations

While we observed that plain RRC and Mixup was sufficient for cloning, we found that better performance and robustness is possible by applying direct pitch shifting and duration scaling to the waveforms. The intuition for this idea is that prosodic variations should not affect speaker identity and thus, using them as augmentations should enable the self-supervised training to better focus on the speaker identity.

Both pitch shifting and duration scaling were implemented via the Praat Tolkit [31]. The amount of shifting and scaling was treated as a hyper-parameter, but in general, it was observed that too much variation was detrimental to the speaker similarity of the cloned utterances. This is presumably because, while speaker identity should not be affected by small shifts in pitch, larger augmentations that take the speaker outside of their normal speaking range will lessen similarity. The same applies to large differences in duration scaling.

It should be noted that these augmentations, unlike RRC, are being done directly in the waveform, and thus precede all the augmentations of the standard BYOL-A training.

2.4.2. External Noise

Since the training of BYOL-a features is general purpose, we expect the acoustic conditions of each utterance to be present in the representations. Since speaker identity is invariant to noises that may exist in the dataset, this is undesirable for our task. In order to make the self supervised features more robust to the acoustic conditions, we utilized the background noise dataset from the Chime-4 challenge [32].

Similarly to [20], the proposed augmentation consists of adding a randomly selected piece of noise with a randomly selected SNR to the waveform. Since the noises of these dataset are sampled from sources which may be present in real audio utterances, we expect this augmentation to better isolate the speaker identity in real world applications.

3. Experiments and Results

3.1. Experimental Setup

The audio features used for pre-training the BYOL-A model are log-scaled mel spectrograms from audios resampled to 16Khz, with 64 mel spaced frequency bins, window size of 64 ms and hop size of 10ms. During training, one second segments are randomly extracted from the audio, with both of the encoder networks operating on the same second. When duration augmentations were used, the beginning of each second is adjusted so that it corresponds to the same point of the waveform in the two augmented views.

The dimension of the BYOL-A embeddings used, is 512. We found that increasing the embedding size beyond that, led to the TTS decoder learning to rely on the embeddings for linguistic information as well as speaker identity, which is undesirable for cloning, as traces of the content of the reference utterance can be heard in the generated voice.

For the prosodic augmentations, a pitch shifting of -1 and 1 semitones was used and a time stretching of 0.95 and 1.05 was found to yield the best results. The external noises were added with an SNR that was randomly sampled from $[5, 10, 25]$. Both of those augmentations, as well as the Gaussian augmentations, when used, were independently done on each augmented view with a probability of 50%.

The TTS architecture follows [33] with a duration predictor and a Gaussian Upsampler to replace the attention mechanism [3]. The pre-trained embeddings are concatenated with the encoder outputs, before they are fed to the predictor and the upsampler. The synthesized features are 22 LPC features, i.e. 20 Bark-scale cepstral coefficients, the pitch period and the pitch correlation, that are then vocoded using the LPCNet Vocoder [27].

For the training of our models, we use the VCTK multispeaker dataset [34] which contains 108 native English speakers, with various accents, both female and male. The total number of the training sentences seen by our model is 44k.

The evaluation of the models was done on an internal crowdsourced multispeaker dataset with 196 native English speakers. For the subjective evaluations, as well as the speaker similarity metric, eight speakers (four female and four male) were randomly selected and two utterances from each speaker were used as a target for cloning. For the MCD metric, we simply randomly selected 100 utterances from the dataset. The normalization of the features before the embedding extraction was done using the statistics of the VCTK dataset, since the actual dataset from which the utterances are drawn from should not need to be available.

As a baseline we used the same non-Attentive Tacotron architecture, but with speaker embeddings extracted using the Deep learning package Resemblyzer [17] trained on the VoxCeleb2 dataset [35] which contains 6,112 speakers and 1.2 M utterances. We also trained from scratch a new Resemblyzer model, only using the VCTK dataset, in order to make the comparison with our models more meaningful. The dimension of the feature embeddings for our baseline models is 256.

Audio samples from our experiments are available at https://innoetics.github.io/publications/ssl-cloning/index.html.

3.2. Objective Evaluation

We adapt the metric s2t-same from [36] which measures how similar synthesized audio from a synthesized speaker is to ground truth audio from the same speaker. While in the original context, this metric was used for speakers of the training dataset, here we use it to measure the speaker fidelity for unseen speakers.

First, we extract the speaker-level d-vectors [37] [18] using the Resemblyzer package [17]. We then compute the average of those d-vectors for each speaker both from the synthesized sentences and the ground truth utterances.

This metric is then defined as:

$$\underset{j}{median}\, d(V_j^s, V_j^t) \tag{4}$$

where the distance d is the cosine distance [38] $d(u_1, u_2) = 1 - \frac{u_1}{\|u_1\|}\frac{u_2}{\|u_2\|}$, V_j^s, V_j^t are the averaged d-vectors over all ground truth utterances of speaker j and of all synthesized sentences from speaker j respectively.

We evaluate this similarity metric with clean and noisy utterances as target sentences. All the noisy utterances were randomly sampled from a (unseen during training) set of noises from the Chime-4 challenge, with an SNR of 5, which was the largest SNR seen during training.

We also use Mel Cepstral Distortion [39] as an additional metric, which measures the similarity of two aligned audio sequences. We therefore compare the similarity of sentences from

Table 1: *Objective Metrics. Lower is better for both.*

Method	s2t-same clean	s2t-same noisy	MCD
d-vectors VoxCeleb2	0.15	0.20	4.05
d-vectors VCTK	0.27	0.29	4.19
Mixup+RRC (BYOL-A)	0.22	0.34	7.49
Mixup+RRC+Pros	0.25	0.35	7.50
Mixup+RRC+Noise	0.23	0.31	7.71
Mixup+RRC+Pros+Noise	0.23	0.28	7.85

an unseen speaker, with the same sentence synthesized from our model given the embedding from the model under evaluation. In order to align the sequences, Dynamic Time Warping (DTW) is used. Only clean utterances were used for this metric.

In Table 1 we can see the results of the objective metrics. The augmentations used for each experiment are shown in the Method column where Pros denotes both of the prosody augmentations defined in 2.4.1, and Noise denotes both the Gaussian and the Chime-4 external augmentations, since early experimentation shown that they work better in tandem.

We see that the d-vectors pre-trained in the VoxCeleb2 outperform the other models, presumably because of their better generalization ability which stems from the fact that they were trained on a much larger dataset. We do observe however that our models all outperform the baseline when trained in the VCTK dataset, at least for the clean utterances.

The better performance of the baseline models when presented with noisy utterances is somewhat surprising. We conjecture that since the training algorithm for these features is based on speaker verification, the encoder learned to ignore all information besides speaker identity, thereby making it robust to noises.

We also note that our augmentations improve upon the vanilla BYOL-A algorithm in terms of the s2t-same metric, although not by much. Additionally, we observe that noise augmentations, when present, lead to less degradation of quality when using noisy utterances as reference, which justifies their inclusion.

The MCD metric deteriorates when we use the extra augmentations. A possible explanation is that the MCD metric is not affected simply by the speaker identity but on other factors as well, such as prosodic information (duration differences for example can induce a high cost on the DTW algorithm). Since we nudged the embeddings to become more independent of those variations, we can explain the deterioration, by the model failing to capture the rest of the speech characteristics, something which is in general desirable for voice cloning. The low values of the baselines for these metric are more troubling and contradict this observation however.

3.3. Subjective Evaluation

We use crowdsourcing to evaluate both the speaker similarity of the synthesized utterance to the target speaker and the overall quality of the speech from a scale from 1 to 5. We have excluded all the test pages with wrong validation utterance scores, with very low (1 or 2) natural speech scores, with the exact same score in all utterances of the page, and with average page score higher than the natural speech score. This filtering process resulted in 187 listeners left to evaluate our samples. Both clean and noisy utterances were used, where the noisy utterances were derived the same way as in Section 3.2.

Table 2: *Subjective results on unseen speakers with 95% confidence intervals. Bold results correspond to the best model for each metric.*

Method	Similarity MOS	Quality MOS
Ground Truth	4.19 ± 0.04	4.27 ± 0.05
d-vectors VoxCeleb2	3.26 ± 0.16	$\mathbf{3.58} \pm 0.10$
d-vectors VCTK	3.21 ± 0.16	3.47 ± 0.14
Mixup+RRC (BYOL-A)	3.15 ± 0.12	3.45 ± 0.10
Mixup+RRC+Pros	$\mathbf{3.30} \pm 0.11$	3.49 ± 0.10
Mixup+RRC+Noise	3.17 ± 0.12	3.47 ± 0.10
Mixup+RRC+Pros+Noise	3.03 ± 0.12	3.48 ± 0.10

The results for regular utterances can be seen in Table 2 and for noisy utterances in Table 3. We can see that both the quality and the speaker similarity of our models is comparable to the d-vectors model trained in the VoxCeleb2 dataset and usually outperforms or is very close to the d-vectors trained on the VCTK dataset. We also outperform both baselines when using the pre-trained model with the prosody augmentations with clear utterances.

The inclusion of noise augmentations is further justified, as they are the best performing of our models when evaluated on the noisy utterances. They also perform quite similarly to the baseline models.

Table 3: *Subjective results on unseen speakers with noisy utterances with 95% confidence intervals. Bold results correspond to the best model for each metric*

Method	Similarity MOS	Quality MOS
Ground Truth	4.19 ± 0.04	4.27 ± 0.05
d-vectors VoxCeleb2	3.03 ± 0.16	$\mathbf{3.52} \pm 0.10$
d-vectors VCTK	$\mathbf{3.14} \pm 0.17$	3.38 ± 0.15
Mixup+RRC (BYOL-A)	2.98 ± 0.13	3.38 ± 0.10
Mixup+RRC+Pros	2.97 ± 0.13	3.37 ± 0.11
Mixup+RRC+Noise	3.04 ± 0.13	3.47 ± 0.10
Mixup+RRC+Pros+Noise	3.12 ± 0.13	3.38 ± 0.10

4. Conclusions

We present a new voice cloning architecture, based on self-supervised features that are pre-trained on an unlabeled dataset. We show that it is close in performance to the baseline of speaker features that are pre-trained on speaker verification tasks even when used on a fraction of the dataset, and even without having any information about the speaker identity of the training sentences. We also further extend the set of augmentations applied in the self-supervised algorithm that both improve the cloning performance and quality and make the model more robust to noise in the target utterances. Future work on this topic could include further exploration of augmentations to improve performance, or the utilization of self supervised features for different TTS tasks such as prosody transfer or emotional speech synthesis.

5. References

[1] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, and S. B. *et al.*, "Tacotron: Towards end-to-end speech synthesis," in *Proc. Interspeech*, 2017, pp. 4006–4010.

[2] J. Shen, R. Pang, R. J. Weiss, and M. S. et al., "Natural tts synthesis by conditioning wavenet on mel spectrogram predictions," in *Proc. ICASSP*. IEEE, 2018, pp. 4779–4783.

[3] J. Shen, Y. Jia, M. Chrzanowski, Y. Zhang, I. Elias, H. Zen, and Y. Wu, "Non-attentive tacotron: Robust and controllable neural tts synthesis including unsupervised duration modeling," *ArXiv*, vol. abs/2010.04301, 2020.

[4] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech: Fast, robust and controllable text to speech," *NeurIPS*, vol. 32, 2019.

[5] W. Ping, K. Peng, A. Gibiansky, S. O. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep voice 3: 2000-speaker neural text-to-speech," in *Proc. ICLR*, 2018.

[6] M. Chen, X. Tan, Y. Ren, J. Xu, H. Sun, S. Zhao, and T. Qin, "Multispeech: Multi-speaker text to speech with transformer," in *Proc. Interspeech*, 2020.

[7] Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Z. Chen, R. Skerry-Ryan, Y. Jia, A. Rosenberg, and B. Ramabhadran, "Learning to speak fluently in a foreign language: Multilingual speech synthesis and cross-language voice cloning," 2019.

[8] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," *NeurIPS*, vol. 31, 2018.

[9] Q. Xie, X. Tian, G. Liu, K. Song, L. Xie, Z. Wu, H. Li, S. Shi, H. Li, F. Hong *et al.*, "The multi-speaker multi-style voice cloning challenge 2021," in *Proc. ICASSP*. IEEE, 2021, pp. 8613–8617.

[10] Y. Taigman, L. Wolf, A. Polyak, and E. Nachmani, "Voiceloop: Voice fitting and synthesis via a phonological loop," *Proc. ICLR*, 2017.

[11] Y. Chen, Y. Assael, B. Shillingford, D. Budden, S. Reed, H. Zen, Q. Wang, L. C. Cobo, A. Trask, B. Laurie, C. Gulcehre, A. van den Oord, O. Vinyals, and N. de Freitas, "Sample efficient adaptive text-to-speech," in *Proc. ICLR*, 2019.

[12] H.-T. Luong and J. Yamagishi, "Nautilus: a versatile voice cloning system," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2967–2981, 2020.

[13] S.-F. Huang, C.-J. Lin, and H.-y. Lee, "Meta-tts: Meta-learning for few-shot speaker adaptive text-to-speech," *arXiv preprint arXiv:2111.04040*, 2021.

[14] G. Maniati, N. Ellinas, K. Markopoulos, G. Vamvoukakis, J. S. Sung, H. Park, A. Chalamandaris, and P. Tsiakoulis, "Cross-Lingual Low Resource Speaker Adaptation Using Phonological Features," in *Proc. Interspeech*, 2021, pp. 1594–1598.

[15] E. Casanova, J. Weber, C. Shulby, A. C. Junior, E. Gölge, and M. A. Ponti, "Yourtts: Towards zero-shot multi-speaker tts and zero-shot voice conversion for everyone," *arXiv preprint arXiv:2112.02418*, 2021.

[16] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *Proc. ICASSP*. IEEE, 2018, pp. 5329–5333.

[17] L. Wan, Q. Wang, A. Papir, and I. Lopez-Moreno, "Generalized end-to-end loss for speaker verification," *Proc. ICASSP*, pp. 4879–4883, 2018.

[18] Y. Jia, Y. Zhang, R. J. Weiss, Q. Wang, J. Shen, F. Ren, Z. Chen, P. Nguyen, R. Pang, I. Lopez-Moreno, and Y. Wu, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *NeurIPS*, 2018.

[19] S. Choi, S. Han, D. Kim, and S. Ha, "Attentron: Few-shot text-to-speech utilizing attention-based variable-length embedding," vol. abs/2005.08484, 2020.

[20] W.-N. Hsu, Y. Zhang, R. J. Weiss, Y.-A. Chung, Y. Wang, Y. Wu, and J. Glass, "Disentangling correlated speaker and noise for speech synthesis via data augmentation and adversarial factorization," in *Proc. ICASSP*, 2019, pp. 5901–5905.

[21] J. Cong, S. Yang, L. Xie, G. Yu, and G. Wan, "Data efficient voice cloning from noisy samples with domain adversarial training," *Proc. Interspeech*, 2020.

[22] L. Wang, P. Luc, Y. Wu, A. Recasens, L. Smaira, A. Brock, A. Jaegle, J.-B. Alayrac, S. Dieleman, J. Carreira *et al.*, "Towards learning universal audio representations," *Proc. Interspeech*, 2021.

[23] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *NeurIPS*, vol. 33, pp. 12 449–12 460, 2020.

[24] J.-B. Grill, F. Strub, F. Altch'e, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. Á. Pires, Z. D. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko, "Bootstrap your own latent: A new approach to self-supervised learning," *NeurIPS*, vol. abs/2006.07733, 2020.

[25] X. Chen and K. He, "Exploring simple siamese representation learning," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15 745–15 753, 2021.

[26] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, "Byol for audio: Self-supervised learning for general-purpose audio representation," in *PRoc. IJCNN*, 2021.

[27] J.-M. Valin and J. Skoglund, "Lpcnet: Improving neural speech synthesis through linear prediction," *Proc. ICASSP*, pp. 5891–5895, 2019.

[28] T. Lillicrap, J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *CoRR*, 09 2015.

[29] Y. Tian, X. Chen, and S. Ganguli, "Understanding self-supervised learning dynamics without contrastive pairs," *ICML*, 2021.

[30] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. ICLR*, 2018.

[31] R. Corretge, "Praat vocal toolkit," 2012-2020. [Online]. Available: http://www.praatvocaltoolkit.com/

[32] 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.

[33] N. Ellinas, G. Vamvoukakis, K. Markopoulos, A. Chalamandaris, G. Maniati, P. Kakoulidis, S. Raptis, J. Sung, H. Park, and P. Tsiakoulis, "High quality streaming speech synthesis with low, sentence-length-independent latency," in *Proc. Interspeech*, 2020.

[34] J. M. K. Veaux, Christophe; Yamagishi, "Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," 2017.

[35] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Science and Language*, 2019.

[36] D. Stanton, M. Shannon, S. Mariooryad, R. J. Skerry-Ryan, E. Battenberg, T. Bagby, and D. Kao, "Speaker generation," *ArXiv*, vol. abs/2111.05095, 2021.

[37] E. Variani, X. Lei, E. McDermott, I. Lopez-Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," *Proc. ICASSP*, pp. 4052–4056, 2014.

[38] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[39] R. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment," *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing*, vol. 1, pp. 125–128 vol.1, 1993.