Speaker verification-derived loss and data augmentation for DNN-based multispeaker speech synthesis

Beáta Lőrincz[1,2], Adriana Stan[1], Mircea Giurgiu[1]

[1]Communications Department, Technical University of Cluj-Napoca, Romania

[2]Computer Science Department, Babeș-Bolyai University of Cluj-Napoca, Romania

{beata.lorincz, adriana.stan, mircea.giurgiu}@com.utcluj.ro

Abstract—**Building multispeaker neural network-based text-to-speech synthesis systems commonly relies on the availability of large amounts of high quality recordings from each speaker and conditioning the training process on the speaker's identity or on a learned representation of it. However, when little data is available from each speaker, or the number of speakers is limited, the multispeaker TTS can be hard to train and will result in poor speaker similarity and naturalness.**

In order to address this issue, we explore two directions: forcing the network to learn a better speaker identity representation by appending an additional loss term; and augmenting the input data pertaining to each speaker using waveform manipulation methods. We show that both methods are efficient when evaluated with both objective and subjective measures. The additional loss term aids the speaker similarity, while the data augmentation improves the intelligibility of the multispeaker TTS system.

Index Terms—**multispeaker text-to-speech synthesis, speaker verification, data augmentation, deep learning, limited training data**

I. Introduction

Following the evolution of fundamental deep neural networks (DNN), the speech synthesis domain achieved results that were previously deemed unimaginable. Text-to-speech (TTS) implementations using recurrent architectures [1, 2], convolutional components [3, 4], transformers [5] or, more recently, the normalizing flow-based concept [6] are currently driving not only the research community, but are also employed within large scale commercial systems. These systems are capable of synthesising speech within the single or multiple speaker identity frameworks.

Multispeaker TTS systems have the advantage of incorporating multiple vocal identities into a single network, and allow the user to select any of these identities without the need to change the model. As a result, the majority of the state-of-the-art synthesis systems also incorporate a multispeaker option. For instance, multispeaker capabilities were appended into two follow-up versions [4, 7] of the Deep Voice system [8]. In [9] Tacotron [1] was extended with a bank of embeddings which learn the speaker identities in an unsupervised manner. MultiSpeech [10] is a transformer based multispeaker TTS system derived from [11]. [12] extends ClariNet [13] to support multispeaker speech synthesis.

The most common approach for integrating multiple speakers in TTS systems is implemented by implicitly or explicitly learning a set of speaker-dependent representations, also called *embeddings*. In [14] the authors compare various neural speaker embeddings and report that learnable dictionary encodings improve the speaker similarity for zero-shot speaker adaptation. [15] presents speaker, language and stress/tone embeddings used for TTS that can synthesize speech in multiple speaker identities and languages. Other techniques employ transfer learning methods to benefit from other speech processing applications, such as speaker verification [16] or speech recognition [17].

Yet, building multispeaker models requires large amounts of high quality speech recordings. When data is not readily available, data augmentation methods have been applied to overcome this barrier. In [18] augmentation methods such as frequency warping, duration and loudness control are used for producing additional speaker data. In [19], TTS systems are trained with data obtained through voice conversion mechanisms. Artificial speaker data and low-quality data usage is experimented within [20] and the authors report improvements on the synthetic speech's naturalness.

One way to alleviate the task of learning speaker-dependent embeddings, as well as the lack of sufficient speaker data is to use a speaker verification (SV) network. [21] examines whether the multispeaker TTS system can improve learning embeddings for the task of speaker verification, while [16] uses the speaker verification system to learn speaker embeddings used to condition the synthesised speech for different speakers. In [22] the authors present a model that uses a speaker verification component to enforce the learning of speaker identities.

Considering the different methods of encoding speaker identities in multispeaker TTS, in this work we propose to enhance the learning of vocal identities by using speaker verification derived metrics and speaker augmented data within the training process. Our work was developed in parallel and shares similarities with [22] in the sense that the feedback of the verification network is incorporated into the model within the loss function. However, the differences are twofold: firstly, we optimize both with cosine similarity and equal error rates calculated by the speaker verification model; secondly, we apply speaker augmentation methods and measure the effect of adding artificially produced data to the training dataset.

II. METHOD OVERVIEW

The main objective of our study is to improve the speaker identity learning process in multispeaker TTS systems when a limited amount of speaker data is available. In order to achieve this, two scenarios are explored: (i) adding an additional loss term obtained from a pre-trained speaker verification network, and (ii) augmenting the data pertaining to each speaker through waveform manipulation methods. The starting point for all the experiments is a text-to-speech synthesis system based on a convolutional architecture with guided attention [3]. The original model does not include speaker identity conditioning, and therefore its architecture was modified to incorporate a learned speaker embedding representation across all of its information channels (see Figure 1). As a preliminary step for all the speaker similarity improvement methods, the original architecture, not conditioned on the speaker identity, was pre-trained with all the available data. The pre-training provides a stable uniform initialisation for the succeeding experiments.

In the first speaker similarity improvement scenario we rely on the information provided by a pre-trained speaker verification (SV) system. The speaker embeddings generated by the SV network from the synthesised output and the embeddings extracted from natural samples are evaluated using the *cosine similarity (CS)* function or the *equal error rate (EER)* measure. The CS value is equal to the batch average of the cosine similarity of the synthesised output with respect to its corresponding natural sample. For EER, depending on the training batch's speaker identities content, a list of random natural utterances are selected and compared with the synthesised output. These speaker verification-based metrics are sequentially appended as additional weighted terms to the overall loss function of the TTS system, and referred to as l_{SV}. The resulting loss function of the TTS system is presented in Eq. 1. α, β and γ were empirically determined during the experiments. An overview of the loss modification process is shown in Figure 1.

$$L = \alpha l_{l1} + \beta l_{attention} + \gamma l_{SV} \qquad (1)$$
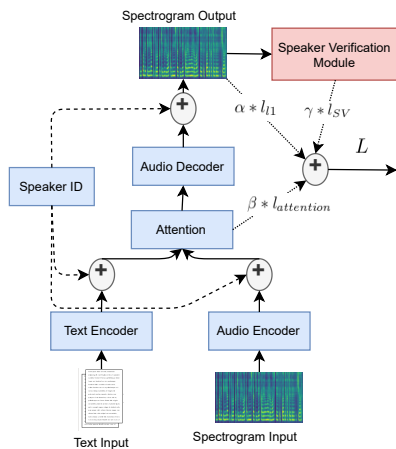

Fig. 1. The model architecture: speaker IDs are appended to several components of the TTS system; the speaker verification network delivers the objective measures used in the additional loss term.

As a result, we obtain a first set of systems for comparison: the baseline system conditioned on the speaker id (B); the baseline system trained with an additional loss term calculated with cosine similarity for the speaker embedding (B+CS); and the baseline system trained with an additional loss term calculated from the EER of the SV network (B+E).

The other improvement scenario relies on the use of basic waveform manipulation methods to augment the data available for each speaker in the training set. The first method is inspired by [20] and uses basic waveform resampling to modify the overall duration of the utterance. In the second method, we employed the Pitch Synchronous Overlap and Add (PSOLA) [23] algorithm–commonly used in unit selection concatenative speech synthesis–to manipulate the duration and pitch of each of the speakers' utterances. As opposed to the simple waveform resampling, PSOLA takes into account the pitch periods and thus yields, to a certain extent, a more natural output. By adding l_{SV} to the TTS system, our main goal was to shift the learning process more towards the speaker identity, even if this meant that it could potentially alter the intelligibility or naturalness of the synthesised speech. The data augmentation methods were combined with the previous loss term computation strategies and yielded additional systems' comparisons.

III. EVALUATION

A. Training data and speaker data augmentation

The training data for our systems consists of the SWARA Romanian multispeaker parallel corpus [24]. It includes 18 speakers: 10 female and 8 male voices, with the number of utterances per speaker being between 1000 and 1500. The data is sampled at 48kHz with 16 bps and was resampled at 16kHz to make the training process faster. The entire SWARA corpus was used to train the speaker verification network on which the additional loss terms are computed.

For the multispeaker TTS systems' training we first selected 3 subsets from the corpus corresponding to various amounts of data for each speaker: ALL, RND1 and RND1-100. A detailed description of these subsets is shown in Table I. In the smaller subsets, parallel utterances were selected for all speakers such that their content does not influence the training and evaluation.

With respect to the data augmentation process, the waveform resampling was performed with the SoX[1] tool, as in [20]. The different resampling ratios are shown in Table I, and the system is referred to as RND1-100-UP-DOWN. The PSOLA-based waveform manipulation used the Pytsmod tool.[2] Three separate modification sets were created: one through duration changes (RND1-100-PSOLA-DUR), another one by warping the F_0 values (RND1-100-PSOLA-F0), and the last one by modifying both the duration and the F_0 values (RND1-100-PSOLA-MIX). Because there is no guarantee

TABLE I

TABLE I
TRAINING DATA SELECTION AND AUGMENTATION OVERVIEW

Data subset ID	Description
ALL (21h:54m)	1000-1500 utterances selected from each speaker
RND1 (10h:45m)	500 utterances selected from each speaker
RND1-100 (2h:24m)	100 utterances selected from each speaker
RND1-100-UP-DOWN (12h:01m)	100 utterances selected from each speaker; each utterance resampled at 0.95, 0.975, 1.025 and 1.05 ratios, resulting in 500 utterances per speaker
RND1-100-PSOLA-DUR (12h:35m)	100 utterances selected from each speaker; for each utterance from a number of 7 time domain augmented files (with ratios 0.85, 0.90, 0.95, 1.05, 1.10, 1.15, 1.20) the best 4 selected, resulting in 500 utterances per speaker
RND1-100-PSOLA-F0 (12h:31m)	100 utterances selected from each speaker; for each utterance from a number of 7 frequency domain augmented files (with ratios 0.70, 0.80, 0.90, 1.05, 1.10, 1.20, 1.50) the best 4 selected, resulting in 500 utterances per speaker
RND1-100-PSOLA-MIX (12h:49m)	100 utterances selected from each speaker; each utterances' duration modified by 1.3 and 0.8, and F_0 by 0.8, 1.2, resulting in 500 utterances per speaker

TABLE II
WER AND **EER** (%) RESULTS FOR THE EVALUATED SYSTEMS

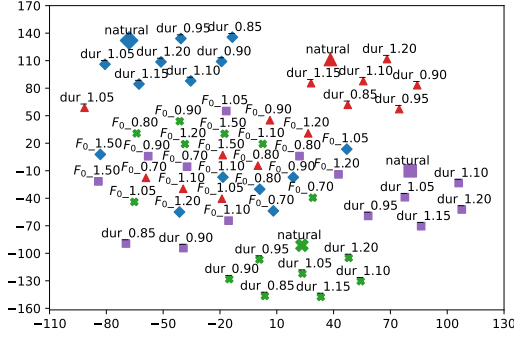|	WER [%]			EER [%]		
	B	**B+CS**	**B+E**	**B**	**B+CS**	**B+E**
ALL	9.54	7.66	8.26	6.94	4.66	4.66
RND1	9.99	8.67	9.86	4.86	4.00	4.66
RND1-100	11.13	10.21	13.26	5.55	5.33	5.33
RND1-100-UP-DOWN	12.42	-	-	8.66	-	-
RND1-100-PSOLA-F0	14.04	15.75	14.18	8.66	10.66	11.33
RND1-100-PSOLA-DUR	11.84	13.62	10.32	8.33	6.25	10.00
RND1-100-PSOLA-MIX	10.05	-	16.00	9.72	-	6.94

Fig. 2. t-SNE plots for speaker embeddings of natural and their corresponding augmented samples. The speakers are color coded, the original natural samples are marked as *natural*, while the augmented versions are marked with *dur* and F_0 followed by the modification ratio.

that the PSOLA modifications do not alter the speaker identity, we used a speaker verification network trained on the VoxCeleb2 dataset [25] to extract the speaker embedding vectors from all the natural and modified speech samples. A t-Distributed Stochastic Neighbour Embedding (t-SNE) [26] visualisation for a subset of them is shown in Figure 2. It can be noticed that the samples with the modified duration are closer to the natural ones, as opposed to the samples where the F_0 is modified. Using the t-SNE representations, in all three modification sets mentioned above, for each natural sample we selected the 4 closest modified samples in terms of Euclidean distance. This ensured that the number of samples available for each speaker in the `RND1-100` scenario is equal to the one in the `RND1` scenario, and their results can be directly compared.

B. Objective evaluation

The starting point TTS system is based on a single speaker implementation[3] of [3] extended to a multispeaker scenario following ideas from a different implementation of the same

[3]https://github.com/tugstugi/pytorch-dc-tts

tool.[4] The TTS network without conditioning on the speaker identity was first pre-trained over 600 epochs. Using the learned weights for initialisation, all the evaluated synthesis systems (`B`, `B+CS`, `B+E`) with the various data selection criteria were trained up to 1000 epochs. One exception was the `RND1-100` system trained over 5000 epochs so as to address the fewer training samples.

The speaker verification network followed the implementation of [27],[5] used only the SWARA corpus for training and was appended to the TTS system architecture (see Figure 1).

The output of the multispeaker TTS systems was evaluated both objectively, in terms of word error rate (WER) and equal error rate (EER), as well as subjectively through listening tests.[6] The WER should estimate the intelligibility of the system and was computed using a general purpose high-quality automatic speech recognition (ASR) system in Romanian [28]. The ASR's WER over the natural samples was 7.43%. Given a well trained SV network, the EER should be able to measure the objective speaker similarity of the synthetic output with respect to the natural samples. Therefore we used a speaker verification network trained on a large corpora–not including SWARA–and published by the authors of [27]. This external SV network ensured that there is no bias or overfitting of the network for the speakers available in the training data. The SV network evaluation over natural samples from each speaker resulted in an EER value of 4%.

From each multispeaker TTS system 8 samples were generated for each of the 18 speakers available in SWARA, using the Griffin-Lim vocoder [29]. The lexical content was the same across all systems and speakers. The resulting 144 samples were then transcribed with the Romanian ASR tool and the WER was computed against the ground truth transcripts. In the EER evaluation, the same 144 synthesised utterances were paired with natural samples pertaining to the same or to a random different speaker identity resulting in 288 pairs. The WER and EER values for the evaluated systems are summarized in Table II. Given the higher values of the baseline systems

[4]https://github.com/CSTR-Edinburgh/ophelia
[5]https://github.com/clovaai/voxceleb_trainer
[6]Audio samples for the presented systems and samples generated with data augmentation methods are available at: https://speech.utcluj.ro/multispeaker_tts/
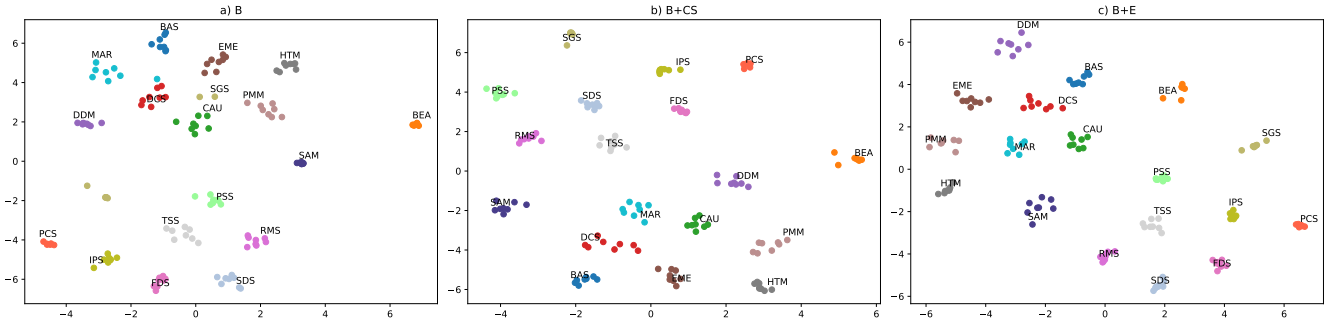
Fig. 3. t-SNE plots of the speaker embeddings extracted by the evaluation speaker verification network from the synthesised outputs of the `ALL` systems in the three training scenarios: `B`, `B+CS` and `B+E`. The speakers are color coded.
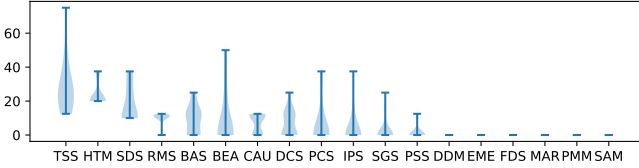

Fig. 4. Violin plots of the EER values obtained by each speaker for all the evaluated TTS systems.

for the `RND1-100-UP-DOWN` and `RND1-100-PSOLA-MIX` datasets, we did not train and evaluate the `B+CS` or `B+E` systems, hence the missing results in Table II.

In terms of objective intelligibility, appending the SV-based loss term, on average, lowers the WER for the systems trained using only the natural speech samples (i.e. `ALL`, `RND1` and `RND1-100`). Better results are obtained when using the cosine similarity measure as opposed to EER. It may be the case that the EER-based loss term is not linearly dependent on the spectrogram output, and therefore does not translate into a well-behaved gradient. On the other hand, the data augmentation seems to affect the intelligibility of the TTS system. The worst effect over the intelligibility is the result of the F_0 manipulation (`RND1-100-PSOLA-F0` - WER: 14.04%). However, it is interesting to notice that when selecting a mix of duration and F_0 manipulated samples (`RND1-100-PSOLA-MIX`) the WER (10.05%) is very close to the one obtained when a similar number of natural samples is used for training (`RND1` - WER: 9.99%). This means, in this case, that a similar intelligibility can be obtained by using only a fifth of the data pertaining to each speaker, and augmenting it with PSOLA-derived samples. With respect to the objective speaker similarity measure, again, the additional loss term (`B+CS` and `B+E`) improves the EER for the natural sample-based systems, with the best performing system being the one which uses the cosine similarity term. However, the training data waveform manipulations does not contribute to a better learning of the speaker identity. This might be explained by the fact that the augmented samples introduce a larger speaker variability and the identity of the speakers in the TTS output is affected by the changes in the duration or F_0 values.

To understand the above results better, we plotted the t-SNE representations of the speaker embeddings extracted by

the evaluation SV network for the systems trained with all the available speaker data (`ALL`) while adding each additional loss term (`CS` or `E`). The results are shown in Figure 3. We can observe that in the case of the baseline system, speakers *SGS* and *MAR* have samples that are further apart in the embedding space, and the distance between these is smaller in case of the `B+CS` and `B+E` systems. While for the *DDM* speaker, the additional terms spread the embeddings through the space. This means that the behaviour of the TTS network favours some of the speakers, while neglecting or averaging the other's characteristics. Further on, we looked at the average EERs for each speaker in the training set over all systems and plotted the violin representations of these values in Figure 4. As it can be easily noticed, there is a clear difference between the learned identities across the speakers. For example *TSS* and *HTM* commonly achieve EERs above 25%, while speakers *SAM*, *PMM* or *MAR* have EERs closer to 0%. We should mention the fact that within this ordering or the t-SNE plots above, there is no definite distinction between the male and female speakers. However this type of ordering and visualisation could provide a more reliable means to select a suitable subset of speaker identities for a multispeaker TTS system.

C. Subjective evaluation

As the objective evaluation of the TTS systems cannot truly reflect the human perception, we also employed a listening test evaluation for a subset of the trained systems. A MUlti
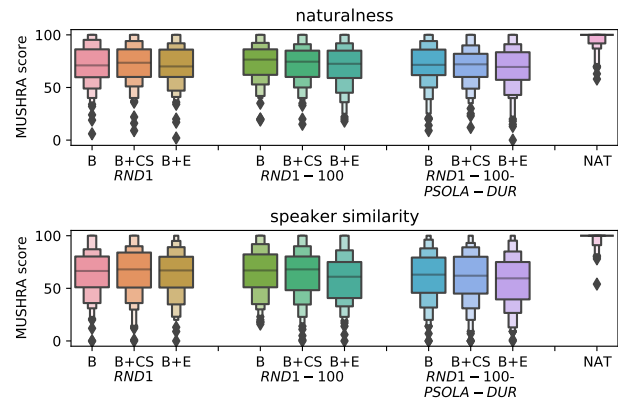

Fig. 5. Letter-value plots of the listening test MuSHRA scores in the naturalness and speaker similarity sections.

Stimulus test with Hidden Reference and Anchor (MuSHRA) method[7] was used to collect the subjective evaluations from 27 native listeners. No lower anchor was used in the test set. Naturalness and speaker similarity results are shown in Figure 5. It can be noticed that although the differences are rather small, the ordering of the systems and improvement methods obtained in the objective evaluation holds true within the listening test as well. This is one additional result supporting the use of high-quality ASR and SV systems in the evaluation of TTS systems' output.

IV. CONCLUSIONS

In this paper we proposed two methods to improve the learning of speaker identities in multispeaker text-to-speech systems. The first one was based on using additional loss terms based on the embedding obtained by a speaker verification network trained on the same data as the TTS system. The second one employed waveform level manipulation methods to augment the available data pertaining to each speaker within the training set. Objective and subjective results showed that the data augmentation can in some cases reduce the required amount of training data to a fifth of the original size, and that the cosine similarity measure increases the intelligibility and speaker similarity of the synthetic output irrespective of the amount of training data available for each speaker. One additional result was the different performance of the TTS system and the improvement methods across the speakers. This leaves space for exploring speaker-dependent training strategies in multispeaker TTS systems.

REFERENCES

[1] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, "Tacotron: Towards end-to-end speech synthesis," *arXiv preprint arXiv:1703.10135*, 2017.

[2] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, "Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[3] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4784–4788.

[4] 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," *Proc. ICLR*, pp. 214–217, 2018.

[5] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T.-Y. Liu, "Fastspeech: Fast, robust and controllable text to speech," in *Advances in Neural Information Processing Systems*, 2019, pp. 3171–3180.

[6] R. Valle, K. Shih, R. Prenger, and B. Catanzaro, "Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis," *arXiv preprint arXiv:2005.05957*, 2020.

[7] A. Gibiansky, S. Arik, G. Diamos, J. Miller, K. Peng, W. Ping, J. Raiman, and Y. Zhou, "Deep voice 2: Multi-speaker neural text-to-speech," in *Advances in neural information processing systems*, 2017, pp. 2962–2970.

[8] S. O. Arik, M. Chrzanowski, A. Coates, G. Diamos, A. Gibiansky, Y. Kang, X. Li, J. Miller, A. Ng, J. Raiman, S. Sengupta, and M. Shoeybi, "Deep voice: Real-time neural text-to-speech," *arXiv preprint arXiv:1702.07825*, 2017.

[9] Y. Wang, D. Stanton, Y. Zhang, R. Skerry-Ryan, E. Battenberg, J. Shor, Y. Xiao, F. Ren, Y. Jia, and R. A. Saurous, "Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis," *arXiv preprint arXiv:1803.09017*, 2018.

[10] M. Chen, X. Tan, Y. Ren, J. Xu, H. Sun, S. Zhao, and T. Qin, "Multispeech: Multi-speaker text to speech with transformer," *arXiv preprint arXiv:2006.04664*, 2020.

[11] 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, 2019, pp. 6706–6713.

[12] J. Park, K. Zhao, K. Peng, and W. Ping, "Multi-speaker end-to-end speech synthesis," *arXiv preprint arXiv:1907.04462*, 2019.

[13] W. Ping, K. Peng, and J. Chen, "Clarinet: Parallel wave generation in end-to-end text-to-speech," *arXiv preprint arXiv:1807.07281*, 2018.

[14] E. Cooper, C.-I. Lai, Y. Yasuda, F. Fang, X. Wang, N. Chen, and J. Yamagishi, "Zero-shot multi-speaker text-to-speech with state-of-the-art neural speaker embeddings," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 6184–6188.

[15] Z. Liu and B. Mak, "Cross-lingual multi-speaker text-to-speech synthesis for voice cloning without using parallel corpus for unseen speakers," *arXiv preprint arXiv:1911.11601*, 2019.

[16] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. Lopez Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," *Advances in neural information processing systems*, vol. 31, pp. 4480–4490, 2018.

[17] K. Inoue, S. Hara, M. Abe, T. Hayashi, R. Yamamoto, and S. Watanabe, "Semi-supervised speaker adaptation for end-to-end speech synthesis with pretrained models," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 7634–7638.

[18] Y. Hwang, H. Cho, H. Yang, D.-O. Won, I. Oh, and S.-W. Lee, "Mel-spectrogram augmentation for sequence to sequence voice conversion," *arXiv preprint arXiv:2001.01401*, 2020.

[19] G. Huybrechts, T. Merritt, G. Comini, B. Perz, R. Shah, and J. Lorenzo-Trueba, "Low-resource expressive text-to-speech using data augmentation," *arXiv preprint arXiv:2011.05707*, 2020.

[20] E. Cooper, C.-I. Lai, Y. Yasuda, and J. Yamagishi, "Can Speaker Augmentation Improve Multi-Speaker End-to-End TTS?" in *Proc. Interspeech 2020*, 2020, pp. 3979–3983.

[21] J. Cho, P. Zelasko, J. Villalba, S. Watanabe, and N. Dehak, "Learning speaker embedding from text-to-speech," *arXiv preprint arXiv:2010.11221*, 2020.

[22] Z. Cai, C. Zhang, and M. Li, "From speaker verification to multispeaker speech synthesis, deep transfer with feedback constraint," *arXiv preprint arXiv:2005.04587*, 2020.

[23] E. Moulines and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech communication*, vol. 9, no. 5-6, pp. 453–467, 1990.

[24] A. Stan, F. Dinescu, C. Țiple, Ș. Meza, B. Orza, M. Chirilă, and M. Giurgiu, "The SWARA speech corpus: A large parallel Romanian read speech dataset," in *2017 International Conference on Speech Technology and Human-Computer Dialogue (SpeD)*. IEEE, 2017, pp. 1–6.

[25] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[26] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[27] J. S. Chung, J. Huh, S. Mun, M. Lee, H. S. Heo, S. Choe, C. Ham, S. Jung, B.-J. Lee, and I. Han, "In defence of metric learning for speaker recognition," *arXiv preprint arXiv:2003.11982*, 2020.

[28] A. Georgescu, H. Cucu, and C. Burileanu, "Kaldi-based DNN Architectures for Speech Recognition in Romanian," in *2019 International Conference on Speech Technology and Human-Computer Dialogue (SpeD)*, 2019, pp. 1–6.

[29] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.

[7]ITU-R Recommendation BS.1534-1