



# SC-GlowTTS: an Efficient Zero-Shot Multi-Speaker Text-To-Speech Model

*Edresson Casanova<sup>1</sup>, Christopher Shulby<sup>2</sup>, Eren Gölge<sup>3</sup>, Nicolas Michael Müller<sup>4</sup>,  
Frederico Santos de Oliveira<sup>5</sup>, Arnaldo Candido Junior<sup>6</sup>, Anderson da Silva Soares<sup>5</sup>,  
Sandra Maria Aluisio<sup>1</sup>, Moacir Antonelli Ponti<sup>1</sup>*

<sup>1</sup> Instituto de Ciências Matemáticas e de Computação, University of São Paulo, Brazil

<sup>2</sup> DefinedCrowd Corp., United States of America

<sup>3</sup> Coqui, Germany

<sup>4</sup> Fraunhofer AISEC, Germany

<sup>5</sup> Federal University of Goiás, Brazil

<sup>6</sup> Federal University of Technology – Paraná, Brazil

edresson@usp.br

## Abstract

In this paper, we propose SC-GlowTTS: an efficient zero-shot multi-speaker text-to-speech model that improves similarity for speakers unseen during training. We propose a speaker-conditional architecture that explores a flow-based decoder that works in a zero-shot scenario. As text encoders, we explore a dilated residual convolutional-based encoder, gated convolutional-based encoder, and transformer-based encoder. Additionally, we have shown that adjusting a GAN-based vocoder for the spectrograms predicted by the TTS model on the training dataset can significantly improve the similarity and speech quality for new speakers. Our model converges using only 11 speakers, reaching state-of-the-art results for similarity with new speakers, as well as high speech quality.

**Index Terms:** zero-shot multi-speaker TTS, text-to-speech, multi-speaker modeling, zero-shot voice conversion.

## 1. Introduction

Text-to-Speech (TTS) systems have received a lot of attention in recent years due to the great advances by deep learning, which have allowed for the popularization of voice applications such as virtual assistants. Most TTS systems were tailored from a single speaker voice, but there is current interest in synthesizing voices for new speakers, not seen during training, employing only a few seconds of speech samples. This approach is called zero-shot multi-speaker TTS (ZS-TTS) as in [1, 2, 3, 4].

ZS-TTS was first proposed [1] by extending the DeepVoice 3 [5]. Also, [2] explored Tacotron 2 [6] using external embeddings extracted from a trained speaker encoder using a generalized end-to-end loss (GE2E) [7], resulting in a model that can generate speech, resembling the target speaker. Similarly, [3] explored Tacotron 2 with different speaker embeddings methods. The authors showed that LDE [8] embeddings improved the similarity and synthesized a more natural speech for novel speakers when compared to X-vector [9] embeddings. The authors in [3] also showed that training a gender-dependent model improves the similarity for unseen speakers.

In this context, a major issue is the similarity gap between observed and unobserved speakers during training. In an attempt to reduce this gap, Attention [4] proposed a fine-grained encoder with an attention mechanism for extracting detailed styles from various reference samples and a coarse-grained encoder. As a result of using several reference samples instead of one, they achieved a better similarity for unseen speakers.

Despite the recent results, zero-shot multi-speaker TTS remains an open problem in particular concerning the difference in the quality of seen and unseen speakers. Also, current approaches rely heavily on Tacotron 2, while there is potential to improve results with the use of flow-based methods [10]. In this context, FlowTron [11] allowed for the manipulation of multiple aspects of speech, such as pitch, tone, speech rate, cadence, and accent. Also, [12] proposed GlowTTS reaching similar quality to Tacotron 2 but with an increase in speed of 15.7 times while permitting speech velocity manipulation.

In this paper, we propose a novel method, Speaker Conditional GlowTTS (SC-GlowTTS), for zero-shot learning of unseen speakers. Our model relies on GlowTTS [12] for the part that converts input characters to spectrograms. SC-GlowTTS uses an external speaker encoder based on Angular Prototypical loss [13], to learn speaker embedding vectors, and adapts the HiFi-GAN [14] vocoder to convert the output spectrograms to the waveform. Our contribution is as follows:

- A novel zero-shot multi-speaker TTS approach that achieves state-of-the-art results with just 11 speakers in the training set;
- An architecture that enables high quality and faster than real-time speech synthesis in the zero-shot multi-speaker TTS setting;
- Adjusting a GAN-based vocoder for the spectrograms predicted by the TTS model on the training dataset, in order to significantly improve the similarity and speech quality for new speakers.

The audio samples for each of our experiments are available on the demo web-site<sup>1</sup>. In addition, for reproducibility the implementation is available at the Coqui TTS<sup>2</sup>, and checkpoints of all experiments are available at the Github repository<sup>3</sup>.

## 2. Speaker Conditional GlowTTS Model

Speaker Conditional Glow-TTS (SC-GlowTTS) builds upon GlowTTS, but includes several novel modifications. In addition to the GlowTTS's transformer-based encoder network, we explore a residual dilated convolutional network [15] and gated convolutional network [16]; to our knowledge, used for the first

<sup>1</sup><https://edresson.github.io/SC-GlowTTS/>

<sup>2</sup><https://github.com/coqui-ai/TTS>

<sup>3</sup><https://github.com/Edresson/SC-GlowTTS>

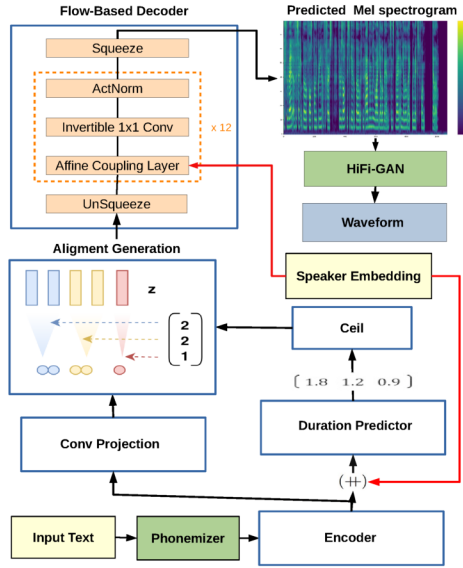


Figure 1: *Speaker Conditional GlowTTS General Architecture.*

time in this context. Our convolutional residual encoder is based on [15], however we used the Mish [17] instead of ReLU activation function. On the other hand, our gated convolutional network [16] consists of 9 convolutional blocks and each block includes a dropout layer, a 1D convolution, and a layer normalization [18]. We use kernel size 5, dilation rate 1, and 192 channels in all convolutional layers. A flow-based decoder is used with the same architecture and configuration as the GlowTTS model. However, to transform it into a zero-shot TTS model, we include speaker embeddings in the affine coupling layers on all 12 decoder blocks. We also used the FastSpeech’s duration predictor network [19] to predict character durations. To capture different speech characteristics of different speakers, we added speaker embeddings to the input of the duration predictor. Finally, the HiFi-GAN [14] is used as a vocoder.

The SC-GlowTTS model, during inference, is illustrated in Figure 1, where (+) indicates concatenation. During training, the model uses the Monotonic Alignment Search (MAS) [12], where the decoder’s objective is to condition the mel spectrogram and an input speaker embedding in a  $P_Z$  prior distribution. The purpose of MAS is to align the  $P_Z$  prior distribution with the encoder’s output. During inference, MAS is not used, instead, the  $P_Z$  prior distribution and alignment are predicted by the text encoder and the duration predictor network. Finally, a latent variable  $Z$  is sampled from the prior distribution  $P_Z$ . Using the inverted decoder and the speaker embeddings, a mel spectrogram is synthesized in parallel, transforming the latent variable  $Z$  via the flow-based decoder.

For brevity, we denominate the SC-GlowTTS model with the transformer, residual convolution, and gated convolution based encoders as SC-GlowTTS-Trans, SC-GlowTTS-Res and SC-GlowTTS-Gated model, respectively.

### 3. Experiments

#### 3.1. Speaker Encoder

Our speaker encoder is a stack of 3 LSTM layers with a linear output layer, similar to [7]. We use 768 LSTM units and 256

units for the linear layer. For training, we used audios sampled at 16 kHz and extracted mel spectrograms using a 1024ms window using the Fast Fourier Transform (FFT), with a hop length of 256 and 1024 FFT components, from which we retain only 80 mel coefficients. Optimization was carried out using the Angular Prototypical [13] loss function different than the original work. The optimizer RAdam [20] was used during 320k steps using 64 speakers per batch, with 10 samples of each speaker and a learning rate of  $10^{-4}$ .

#### 3.2. Zero-Shot Multi-Speaker Tacotron 2

We compare our approach with Tacotron 2. Following the proposal of [2], [21] and [3] we use local sensitive attention [6]. We concatenate the speaker embeddings to the input of the attention module as in [2, 3], given that the latter showed this was adequate for a gender-independent Tacotron model. To alleviate possible issues in the attention module we use Double Decoder Consistency (DDC) [22] with gradual training [23] and guided attention [24]. In Tacotron, the number of output frames per decoder iteration is called the reduction rate ( $R$ ) [23, 6]. The idea of the DDC is to combine two decoders with different reduction factors. One decoder (coarse) works with a higher  $R$  and another decoder (fine) works with a smaller  $R$  value. Gradual training simply starts training with a larger  $R$  and decreases it during the training. In our experiments, we use  $R = 7$  for the coarse decoder, while for the fine decoder we used gradual training, starting from  $R = 7$  and decreasing it as follows:  $R = 5$  at step 10k;  $R = 3$  at step 25k;  $R = 2$  at step 70k.

#### 3.3. Audio datasets

Our speaker encoder was trained with all partitions of the LibriSpeech dataset [25], the English version of Common Voice [26], the VCTK and VoxCeleb (v1 and v2) datasets [27], totaling approximately 25k speakers.

Our zero-shot multi-speaker TTS model is trained using VCTK [28] dataset, an English language dataset containing 44 hours of speech and 109 speakers, sampled at 48KHz. Each speaker pronounces approximately 400 sentences. Pre-processing was carried out in order to remove long periods of silence. We applied voice activity detection (VAD) using Webrtctvad toolkit<sup>4</sup>. We have divided the VCTK dataset into: train, validation (containing the same speakers as the train set) and test. For the test set, we selected 11 speakers not present in the validation or training set; following the proposal by [2], we selected 1 representative from each accent totaling 7F/4M (speakers 225, 234, 238, 245, 248, 261, 294, 302, 326, 335 and 347). For the HiFi-GAN [14] vocoder initial training uses *train-clean-100* and *train-clean-360* partitions of the LibriTTS [29] dataset.

#### 3.4. Experimental setup

We carried out four training experiments:

- **Experiment 1:** Tacotron zero-shot model, described in Section 3.2, trained for 210k steps.
- **Experiment 2:** SC-GlowTTS-Trans model trained for 150k steps.
- **Experiment 3:** SC-GlowTTS-Res model trained for 150k steps.
- **Experiment 4:** SC-GlowTTS-Gated model trained for 150k steps.

<sup>4</sup><https://github.com/wiseman/py-webrtctvad>

In all experiments, we used RAdam [20] with batch size 128, an initial learning rate of  $10^{-3}$ , and Noam’s learning rate schedule [30] with 4000 warmup steps. We use the same configuration to extract the mel spectrograms from the speaker encoder, detailed in the Section 3.1 but with 22khz sampling rate. We use the VCTK dataset in all our experiments using the training, validation, and test partitions as specified in Section 3.3 and we use the validation set to choose the best checkpoint for each experiment comparing the loss value.

In all experiments, we choose to use phonemes as input instead of text. Specifically, we used the Phonemizer tool<sup>5</sup>, which supports several languages. In addition, we add a blank token between each of the phonemes in the input sentence for the GlowTTS-based models, as suggested by the original work<sup>6</sup> [12].

HiFi-GAN v2 model was used as a vocoder, due to its effective speed/quality trade-off. As a starting point, we used the model provided by the authors trained for 500k steps with the LJ Speech [31] dataset. We first trained the HiFi-GAN model for 75k steps with the LibriTTS dataset. Afterward, the model is adjusted for other 190k steps using the VCTK dataset, using the training and validation partitions as specified in Section 3.3.

[14] showed that adjusting the HiFi-GAN model with the spectrogram of the TTS model, improves quality for a single speaker. However, it remains an open question whether it improves; (i) the quality in multi-speaker, (ii) speech similarity for unseen speakers in ZS-TTS settings. To answer this, our TTS models synthesize each of the sentences in the training and validation splits of VCTK dataset. We enabled teacher forcing to keep the alignments between predicted spectrogram frames and the input phonemes. For SC-GlowTTS we use the MAS to align the decoder output with the encoder output. Using these spectrograms extracted from each model, we fine-tuned the checkpoint initially trained with the LibriTTS dataset, for an additional 190k steps, producing the fine-tuned HiFi-GAN (HiFi-GAN-FT).

## 4. Results and Discussion

In this paper, the synthesized speech quality is evaluated using mean opinion score (MOS) study, following [32]. MOS scores were obtained with rigorous crowdsourcing [33]. For the MOS calculation, 15 professional collaborators per audio were invited from a total of 68 unique contributors (35F/33M). To compare the similarity between the synthesized voice and the original speaker, we calculate the Speaker Encoder Cosine Similarity (SECS). The SECS consists of calculating the cosine similarity between the embeddings of two audios extracted from the speaker encoder. It ranges from -1 to 1, and a larger value indicates a stronger similarity [3]. Following [4], we compute SECS using the speaker encoder of the Resemblyzer [7, 34] package; thus, allowing comparison with those studies. We also report the MOS similarity (Sim-MOS) following the work of [2] and [4].

We also compare the run-time of each model by calculating the Real Time Factor (RTF) on a CPU and GPU. For speed tests we used a machine with an NVIDIA GeForce GTX Titan V GPU, an Intel (R) Xeon (R) CPU E5-2603 v4 @ 1.70GHz processor with 6 CPU cores and 15 Gb of RAM. The training was carried out on an NVIDIA V100 GPU. Also, RTF was calculated considering the full synthesis run, from the input

phonemes to the output waveform. We synthesize 15 different sentences as in [35] 10 times for each of the 11 speakers of the VCTK test set and calculated the average.

As a reference sample for the extraction of speaker embeddings, we use the fifth sentence of the VCTK (i.e. speakerID\_005.txt), since all test speakers uttered it and because it is a long sentence (20 words). In this way, all speakers are presented in the zero-shot multi-speaker TTS model by a reference sample with the same number of words and speech content.

For the calculation of MOS and SECS we randomly drew 55 sentences from the *test-clean* subset of the LibriTTS, considering only sentences with more than 20 words. We randomly select five sentences for each of the 11 test speakers, ensuring that all 55 test sentences are synthesized and that all the test speakers are considered. As ground truth, we select 5 audios randomly for each of the 11 test speakers (55 in total), only audios with more than 20 words are studied.

On the other hand, for the SECS ground truth, we compared the 55 audios chosen at random (explained above) with the reference audios used to synthesize the sentences (fifth sentence of the VCTK dataset for each of the test speakers).

Table 1 shows the RTF in CPU and GPU, MOS with 95% confidence intervals and SECS for all of our experiments. Speed tests show that the fastest model on both CPU and GPU is SC-GlowTTS-Gated, followed by the SC-GlowTTS-Res model. The SC-GlowTTS-Trans model is the slowest of the SC-GlowTTS family, however, still much faster than Tacotron 2. Despite this, with the integration with the HiFi-GAN vocoder all models are real time in both CPU and GPU.

SECS score of the ground truth reached 0.9222 because it compares the sample used as a reference with other real speech samples of the same speaker. This value is intended to show an upper bound for SECS, i.e., a model that perfectly “copies” the voice of a target speaker.

The best SECS for synthesis with the HiFi-GAN vocoder (without fine-tuning) was obtained by the SC-GlowTTS-Trans model (experiment 2), followed by Tacotron 2 (experiment 1). The SC-GlowTTS-Res model (experiment 3) achieved the third-best SECS being only better than SC-GlowTTS-Gated (experiment 2). Using the HiFi-GAN-FT, the SC-GlowTTS-Trans model also obtained the best SECS, followed by the SC-GlowTTS-Res model. The SC-GlowTTS-Gated model reached the third-best SECS being only better than the Tacotron 2 model. We found that the fine-tuning of the HiFi-GAN vocoder in the spectrograms extracted from the TTS models significantly improves SECS for the new speakers. The SECS increased from 0.7589 to 0.7791, 0.7641 to 0.8046, 0.7440 to 0.7969 and 0.7432 to 0.7849, respectively, for the models Tacotron 2, SC-GlowTTS-Trans, SC-GlowTTS-Res and SC-GlowTTS-Gated.

For Sim-MOS the results are similar to those of SECS. However, there are some differences which can be explained by the overlapping of the Sim-MOS confidence intervals between the experiments. Improvement with the use of HiFi-GAN-FT can also be seen in all experiments.

Finally, we compare our results with those presented by the Attention model. In [4], the authors reported SECS values, also calculated by the speaker encoder that we use. Although the authors use only 8 speakers (4F/4M) for the test and we use 11 speakers, we believe the comparison is fair and leaves no selection criteria undefined. The Attention model in zero-shot mode reached a SECS of only 0.731. Such a model uses multiple samples to synthesize speech instead of just one, performing few-shot TTS with 8 reference samples. This approach achieves a SECS of 0.788, slightly lower than our best SECS,

<sup>5</sup><https://github.com/bootphon/phonemizer/>

<sup>6</sup><https://github.com/jaywalnut310/glow-tts>

Table 1: Real Time Factor, MOS and Sim-MOS with 95% confidence intervals and the SECS for all our experiments.

Experiment - Model	Vocoder	RTF (CPU - GPU)	SECS	MOS	Sim-MOS
Ground Truth	–	–	0.9236	4.12 $\pm$ 0.06	4.127 $\pm$ 0.06
Attentron ZS [4]	WaveRNN	–	(0.731)	(3.86 $\pm$ 0.05)	(3.30 $\pm$ 0.06)
1 - Tacotron 2	HiFi-GAN	0.5782 - 0.2485	0.7589	3.57 $\pm$ 0.08	3.867 $\pm$ 0.08
	HiFi-GAN-FT	-	0.7791	3.74 $\pm$ 0.08	3.951 $\pm$ 0.07
2 - SC-GlowTTS-Trans	HiFi-GAN	0.3612 - 0.1557	0.7641	3.65 $\pm$ 0.07	3.905 $\pm$ 0.07
	HiFi-GAN-FT	-	<b>0.8046</b>	3.78 $\pm$ 0.07	<b>3.999 <math>\pm</math> 0.07</b>
3 - SC-GlowTTS-Res	HiFi-GAN	0.3597 - 0.1545	0.7440	3.45 $\pm$ 0.09	3.828 $\pm$ 0.08
	HiFi-GAN-FT	-	0.7969	3.70 $\pm$ 0.07	3.916 $\pm$ 0.07
4 - SC-GlowTTS-Gated	HiFi-GAN	0.3474 - 0.1437	0.7432	3.55 $\pm$ 0.08	3.852 $\pm$ 0.08
	HiFi-GAN-FT	-	0.7849	<b>3.82 <math>\pm</math> 0.07</b>	3.952 $\pm$ 0.07

0.8046. Despite the advantage of the few-shot approach, our model still achieves a higher SECS than Attentron. The authors also reported the Sim-MOS reaching  $4.83 \pm 0.02$  for ground truth speech, zero-shot mode Attentron reaches  $3.30 \pm 0.06$  and few-shot mode  $3.57 \pm 0.05$ . Considering these values, our best model was superior at  $3.999 \pm 0.07$ . Furthermore, the results of our model are closer to the ground truth being only 0.128 smaller while in [4] the best experiment’s difference is 1.26.

For the MOS, ground truth speech reached 4.12. The SC-GlowTTS-Gated model with the HiFi-GAN-FT vocoder was the closest, reaching a MOS of 3.82. Moreover, as in SECS, where the HiFi-GAN-FT vocoder improved speech similarity, the best MOS was achieved using the same vocoder. With the adjustment of the HiFi-GAN vocoder in the spectrograms extracted from the TTS model, the MOS for new speakers increased significantly from 3.57 to 3.74, 3.65 to 3.78, 3.45 to 3.70, 3.55 to 3.82, respectively, for all models Tacotron 2, SC-GlowTTS-Trans, SC-GlowTTS-Res and SC-GlowTTS-Gated. Our MOS values are on par with the other state-of-the-art ZS-TTS models such as [3, 4].

## 5. SC-GlowTTS performance with few speakers

To emulate a scenario with few speakers, we reflect our test set by selecting a subset of the VCTK dataset training set. This new training set consists of 11 speakers, 7F/4M. We selected 1 representative for each accent, except for the “New Zealand” accent that has only one speaker and it is in our test set, so we added an “American” speaker instead, the chosen speakers were 229, 249, 293, 313, 301, 374, 304, 316, 251, 297 and 323. From this new training set, we have selected random samples to use as a validation set. As a test set, we use the same one defined in Section 3.3.

We use the SC-GlowTTS-Trans model and train it with the LJSpeech [31] dataset for 290k steps. This pre-training in a single-speaker dataset was carried out to prime the encoder of the model in a larger vocabulary. We fine-tuned the SC-GlowTTS-Trans model in the new training set with only 11 speakers for 70k steps and using the validation set, we selected the best checkpoint as step 66k. In addition, using the HiFi-GAN model trained in the LibriTTS dataset for 75k steps, we adjusted for other 95k steps using the same technique. This new experiment resulted in a SECS of 0.7707, a MOS of  $3.71 \pm 0.07$  and Sim-MOS of  $3.93 \pm 0.08$ . These results are compatible with SECS of 0.7791, MOS 3.74 and Sim-MOS  $3.951 \pm 0.07$  achieved by Tacotron 2, which used a much larger set of 98 speakers. Therefore, our SC-GlowTTS-Trans model converges

with a 8.9 times smaller dataset, with comparable performance to Tacotron 2. We believe that this is an important step forward especially for ZS-TTS in low-resource languages.

## 6. Zero-Shot Voice Conversion

As in the original GlowTTS [12] model, we do not provide any information about the speaker’s identity to the model encoder, so the distribution predicted by the encoder is forced to be independent of the speaker identities. Therefore, like GlowTTS, SC-GlowTTS can convert voices using only the model’s decoder. However, in our work, we condition SC-GlowTTS with external speaker embeddings. It enables our model to resemble the voice for speakers not seen in the training by performing a zero-shot voice conversion. Samples of the zero-shot voice conversion are present on the demo page<sup>7</sup>.

## 7. Conclusions and future work

In this work, we present a novel method, SC-GlowTTS, achieving state-of-the-art ZS-TTS results. We explored three different encoders for the SC-GlowTTS model and showed that a transformer-based encoder gave the best similarity for speakers not seen in the training. Our SC-GlowTTS models are superior to Tacotron 2. Also, when combined with an external speaker encoder, SC-GlowTTS models can perform ZS-TTS with only 11 speakers in the training set. Finally, we found that the adjustment of the HiFi-GAN vocoder in the spectrograms predicted by the TTS model in the training and validation set can significantly improve the similarity and the quality of the synthesized speech (MOS) for speakers not seen in the training. As future work, following the work of [4], we intend to extend the SC-GlowTTS as a few-shot approach.

## 8. Acknowledgements

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. Also, we would like to thank the CyberLabs<sup>8</sup> for financial support for this paper and Defined-Crowd for making industrial-level MOS testing so easily available. Finally, we would like to thank all contributors to the Coqui TTS repository<sup>9</sup>, this work was only possible thanks to the commitment of all.

<sup>7</sup><https://edresson.github.io/SC-GlowTTS/>

<sup>8</sup><https://cyberlabs.ai/>

<sup>9</sup><https://github.com/coqui-ai/TTS>

## 9. References

- [1] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," in *Advances in Neural Information Processing Systems*, 2018, pp. 10019–10029.
- [2] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. L. Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in neural information processing systems*, 2018, pp. 4480–4490.
- [3] 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.
- [4] S. Choi, S. Han, D. Kim, and S. Ha, "Attentron: Few-shot text-to-speech utilizing attention-based variable-length embedding," *arXiv preprint arXiv:2005.08484*, 2020.
- [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," *arXiv preprint arXiv:1710.07654*, 2017.
- [6] 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.
- [7] L. Wan, Q. Wang, A. Papir, and I. L. Moreno, "Generalized end-to-end loss for speaker verification," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4879–4883.
- [8] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," *arXiv preprint arXiv:1804.05160*, 2018.
- [9] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.
- [10] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling, "Improved variational inference with inverse autoregressive flow," in *Advances in neural information processing systems*, 2016, pp. 4743–4751.
- [11] 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.
- [12] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-tts: A generative flow for text-to-speech via monotonic alignment search," *arXiv preprint arXiv:2005.11129*, 2020.
- [13] 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," in *Interspeech*, 2020.
- [14] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *arXiv preprint arXiv:2010.05646*, 2020.
- [15] J. Vainer and O. Dušek, "Speedyspeech: Efficient neural speech synthesis," *arXiv preprint arXiv:2008.03802*, 2020.
- [16] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *International conference on machine learning*. PMLR, 2017, pp. 933–941.
- [17] D. Misra, "Mish: A self regularized non-monotonic neural activation function," *arXiv preprint arXiv:1908.08681*, 2019.
- [18] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.
- [19] 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.
- [20] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the variance of the adaptive learning rate and beyond," *arXiv preprint arXiv:1908.03265*, 2019.
- [21] D. Paul, Y. Pantazis, and Y. Stylianou, "Speaker conditional wavernn: Towards universal neural vocoder for unseen speaker and recording conditions," *arXiv preprint arXiv:2008.05289*, 2020.
- [22] E. Gölge. (2020) Solving attention problems of tts models with double decoder consistency. [Online]. Available: <https://erogol.com/solving-attention-problems-of-tts-models-with-double-decoder-consistency/>
- [23] —. (2019) Gradual training with tacotron for faster convergence. [Online]. Available: <https://erogol.com/gradual-training-with-tacotron-for-faster-convergence/>
- [24] H. Tachibana, K. Uenoyama, and S. Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," *arXiv preprint arXiv:1710.08969*, 2017.
- [25] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5206–5210.
- [26] R. Ardila, M. Branson, K. Davis, M. Henretty, M. Kohler, J. Meyer, R. Morais, L. Saunders, F. M. Tyers, and G. Weber, "Common voice: A massively-multilingual speech corpus," *arXiv preprint arXiv:1912.06670*, 2019.
- [27] J. S. Chung, A. Nagrani, and A. Zisserman, "Vox-celeb2: Deep speaker recognition," in *Proc. Interspeech 2018*, 2018, pp. 1086–1090. [Online]. Available: <http://dx.doi.org/10.21437/Interspeech.2018-1929>
- [28] C. Veaux, J. Yamagishi, K. MacDonald *et al.*, "Superseded-cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit," *University of Edinburgh. The Centre for Speech Technology Research (CSTR)*, 2016.
- [29] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "Libritts: A corpus derived from librispeech for text-to-speech," *arXiv preprint arXiv:1904.02882*, 2019.
- [30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.
- [31] K. Ito *et al.*, "The lj speech dataset," 2017.
- [32] F. Ribeiro, D. Florêncio, C. Zhang, and M. Seltzer, "Crowdmoss: An approach for crowdsourcing mean opinion score studies," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2416–2419.
- [33] Mos testing. DefinedCrowd Corp. [Online]. Available: <https://www.definedcrowd.com/evaluation-of-experience/>
- [34] C. Jemine *et al.*, "Master thesis: Real-time voice cloning," 2019.
- [35] K. Peng, W. Ping, Z. Song, and K. Zhao, "Non-autoregressive neural text-to-speech," in *International Conference on Machine Learning*. PMLR, 2020, pp. 7586–7598.