ONE TTS ALIGNMENT TO RULE THEM ALL

Rohan Badlani, Adrian Łańcucki, Kevin J. Shih, Rafael Valle, Wei Ping, Bryan Catanzaro

{rbadlani, alancucki, kshih, rafaelvalle, wping, bcatanzaro}@nvidia.com
NVIDIA, Santa Clara

ABSTRACT

Speech-to-text alignment is a critical component of neural text-to-speech (TTS) models. Autoregressive TTS models typically use an attention mechanism to learn these alignments on-line. However, these alignments tend to be brittle and often fail to generalize to long utterances and out-of-domain text, leading to missing or repeating words. Most non-autoregressive end-to-end TTS models rely on durations extracted from external sources. In this paper we leverage the alignment mechanism proposed in RAD-TTS and demonstrate its applicability to wide variety of neural TTS models. The alignment learning framework combines the forward-sum algorithm, Viterbi algorithm, and an efficient static prior. In our experiments, the framework improves all tested TTS architectures, both autoregressive (Flowtron, Tacotron 2) and non-autoregressive (FastPitch, FastSpeech 2, RAD-TTS). Specifically, it improves alignment convergence speed, simplifies the training pipeline by eliminating need for external aligners, enhances robustness to errors on long utterances and improves the perceived speech synthesis quality, as judged by human evaluators.

Index Terms: neural speech synthesis, speech text alignments

1. INTRODUCTION

Neural text-to-speech (TTS) models, especially autoregressive TTS models, produce naturally sounding speech for in-domain text [1–3]. However, these models can suffer from pronunciation issues such as missing and repeated words for out-of-domain text, especially in long utterances. A typical neural TTS model consists of an encoder that maps text inputs to hidden states, a decoder that generates mel-spectograms or waveforms from the hidden states, and an alignment mechanism or a duration source that maps the encoder states to decoder inputs [1–7]. Autoregressive TTS models rely on the attention mechanism [8, 9] to align text and speech, typically using a content based attention mechanism [1, 3]. Although recent works have improved alignments by using both content and location sensitive attention [2], such models still suffer from alignment problems on long utterances [6].

In contrast, parallel (non-autoregressive) TTS models factor out durations from the decoding process, thereby requiring durations as input for each token. These models generally rely on external aligners [4] like the Montreal Forced Aligner (MFA) [10], or on durations extracted from a pre-trained autoregressive model(forced aligner) [5, 7, 11] like Tacotron 2 [2]. In addition to the dependency on external alignments, these models can suffer from poor training efficiency, require carefully engineered training schedules to prevent unstable learning, and may be difficult to extend to new languages either because pre-existing aligners are unavailable or their output does not exactly fit the desired format. Ideally, we would like the alignment to be trained end-to-end as part of the TTS model to significantly simplify the training pipeline. We would also like the alignments to converge rapidly as the rest of the TTS pipeline depends on it. Most importantly, the output quality should be better (at least no worse) than if we were to train on alignments provided by external sources.

This work leverages the alignment framework proposed in RAD-TTS [12] to simplify alignment learning in several TTS models[1]. We demonstrate its ability to convert all TTS models to a simpler end-to-end alignment pipeline with better convergence rates and improved robustness to long utterances. We improve prior work on alignments in autoregressive TTS systems [1–3] by adding a constraint that directly maximizes the likelihood of text given speech mel-spectrograms. We demonstrate that this approach can also be used to learn alignments online in parallel TTS models [4, 7, 12], eliminating the need for external aligners. In addition, we further examine the effect of a simple static alignment prior for guiding alignment attention learning [12, 13]. In summary, our results[2] show that TTS models trained with our alignment learning framework have fewer repeated and missing words during inference, improved stability on long sequence synthesis, and improved overall speech quality based on human evaluation.

2. ALIGNMENT LEARNING FRAMEWORK

We extend the alignment learning approach proposed in RAD-TTS [12] to be more broadly applicable to various TTS models, especially autoregressive models. Our alignment framework is presented in Figure 1. It takes the encoded text input $\Phi \in \mathbb{R}^{C_{txt} \times N}$ and aligns it to mel-spectrograms $X \in \mathbb{R}^{C_{mel} \times T}$ where T is number of mel frames and N is the text length.

2.1. Unsupervised alignment learning objective

To learn the alignment between mel-spectrograms(X) and text(Φ), we use the alignment learning objective proposed in

[1] Alignment framework source code is available for Flowtron and FastPitch
[2] Samples available at https://nv-adlr.github.io/one-tts-alignment.

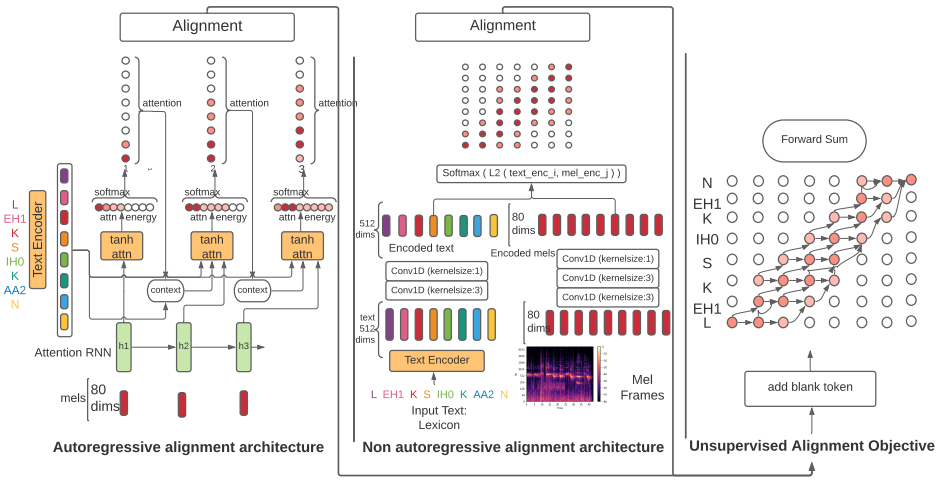


Fig. 1: Overview of the alignment learning: autoregressive models use sequential attention mechanism to generate alignments between text and mels. Parallel models encode text and mel using 1D convolutions and use pairwise L_2 distance to compute alignments. Alignments represent the distribution $P(s_t|x_t)$ used in alignment objective (Eq. 1).

RAD-TTS [12]. This objective maximizes the likelihood of text given mel-spectrograms using the forward-sum algorithm used in Hidden Markov Models [14]. In our formulation, we constrain the alignment between text and speech to be monotonic, in order to avoid missing or repeating tokens. The following equation summarizes the conditional likelihood of text:

$$P\left(S(\Phi) \mid X;\theta\right) = \sum_{\mathbf{s}\in S(\Phi)} \prod_{t=1}^{T} P(s_t \mid x_t;\theta) \tag{1}$$

where s is a specific alignment between mels and text (eg: $s1 = \phi_1, s2 = \phi_1, s3 = \phi_2, \ldots, sT = \phi_N$), $S(\Phi)$ is the set of all possible valid monotonic alignments, $P(s_t|x_t)$ is the likelihood of a specific text token $s_t = \phi_i$ aligned for mel frame x_t at timestep t. It is important to note that the above formulation of the alignment learning objective is applicable to both autoregressive and parallel models. We define the forward sum objective that maximizes (1) as $\mathcal{L}_{ForwardSum}$. Following RAD-TTS, we use the efficient, off-the-shelf CTC [15] implementation from PyTorch to compute this objective.

2.2. Autoregressive TTS Models

Autoregressive TTS models typically use a sequential attention mechanism to learn online alignments. TTS models such as Tacotron [1] and Flowtron [3] use a content based attention mechanism that relies only on decoder inputs and the current attention hidden state to compute an attention map between encoder and decoder steps. Other autoregressive models use a location relative attention mechanism [16] to promote forward movement of alignments [2]. Although alignment learning in these autoregressive models is tightly coupled with the decoder and can be learned with the mel-spectrogram reconstruction objective, it has been observed that the likelihood of a misstep in the alignment increases with the length of the utterance. This results in catastrophic failure on long sequences and out-of-domain text [17]. The application of the unsupervised objective described in Sec 2.1 improves both convergence speed during training and robustness during inference.

Our autoregressive setup uses the standard stateful content based attention mechanism for Flowtron [3] and a hybrid attention mechanism with both content and location based features for Tacotron2 [2]. The location sensitive term (Eq. 4) uses features computed from attention weights at previous decoder timesteps. We use the Tacotron2 encoder to obtain the sequence of encoded text representations $(\phi_i^{enc})_{i=1}^{N}$ and an attention RNN to produce a sequence of hidden states h_t. A simple architecture is used to compute the alignment energies $e_{t,i}$ for text token s_i at timestep t for mel x_t using the tanh attention [9]. The attention weights are computed with softmax over the text domain using the alignment energies. The following equations summarize the attention mechanism:

$$(h_t)_{t=1}^{T} = \text{RNN}(h_{t-1}, x_{t-1}, c_{t-1}) \tag{2}$$

$$c_t = \sum \alpha_{t,i}\phi_i^{enc} \tag{3}$$

$$f_t = F(\alpha_{t-1}) \tag{4}$$

$$e_{t,i} = -v^T \tanh(Wh_t + V\phi_i^{enc} + Uf_{t,i}) \tag{5}$$

$$P(s_t = \phi_i|x_t) = \alpha_{t,i} = Softmax(-e_t)_i, \tag{6}$$

where f_t is the location relative term for location sensitive attention F (cumulative attention from [2] using a concatenation of the attention weights from the previous timestep and the cumulative attention weights). The attention weights model the distribution $P(s_t = \phi_i|x_t)$, which is exactly the right-most term in Eq 1, and we incorporate it as the alignment loss:

$$\mathcal{L}_{align} = \mathcal{L}_{ForwardSum}. \tag{7}$$

2.3. Parallel TTS Models

As parallel TTS models have durations factored out from the decoder, the alignment learning module can be decoupled from the mel decoder as a standalone aligner. This provides a lot of flexibility in choosing the architecture to formulate the distribution $P(s_t|x_t)$, where s_t is a random variable for a text token aligned at timestep t for mel frame x_t. Similar to GlowTTS [6] and RAD-TTS [12], we compute the soft alignment distribution based on the learned pairwise affinity between all text tokens and mel frames, which is normalized with softmax across the text domain:

$$D_{i,j} = dist_{L2}(\phi_i^{enc}, x_j^{enc}), \tag{8}$$

$$\mathcal{A}_{soft} = \texttt{softmax}(-D, \texttt{dim} = 0). \tag{9}$$

We use two simple convolutional encoders from RAD-TTS [12] for encoding text Φ as Φ^{enc} and mel-spectograms X as X^{enc} with 2 and 3 1D convolution layers respectively. In Section 3, we demonstrate that the same architecture works well with different parallel TTS models such as FastPitch and FastSpeech2. Parallel models require alignments to be specified beforehand, typically in the form of the number of output frames for every input phoneme, equivalent to a binary alignment map. However, attention models produce soft alignment maps, constituting a train-test domain gap. Following [6, 12], we use Viterbi algorithm to find the most likely monotonic path through the soft alignment map in order to convert soft alignments($\mathcal{A}_{soft}$) to hard alignments($\mathcal{A}_{hard}$). We further close the gap between soft and hard alignments by forcing $\mathcal{A}_{soft}$ to match $\mathcal{A}_{hard}$ as much as possible by minimizing their KL-divergence(L_{bin}):

$$\mathcal{L}_{bin} = \mathcal{A}_{hard} \odot \log \mathcal{A}_{soft}, \tag{10}$$

$$\mathcal{L}_{align} = \mathcal{L}_{ForwardSum} + \mathcal{L}_{bin}. \tag{11}$$

where $\odot$ is Hadamard product, L_{align} is final alignment loss.

2.4. Alignment Acceleration

Faster convergence of alignments means faster training of the TTS model, as training the decoder needs a stable alignment. The length of mel-spectrograms is known upfront during training, hence we use a static 2D prior [12], that is wider near the center and narrower near the corners to accelerate the alignment by making far-off-diagonal elements less probable. This idea has been explored by Tachibana et al [13] by introducing a new loss promoting near-diagonal alignments. We believe auxiliary loss [13] should yield similar results to our static prior approach. We apply the prior (f_B) over the alignment ($P(s \mid X{=}x_t)$) to obtain the following posterior:

$$f_B(k, \alpha, \beta) = \binom{N}{k} \frac{B(k+\alpha)B(N-k+\beta)}{B(\alpha, \beta)} \tag{12}$$

$$P_{posterior}(\Phi{=}\phi_k \mid X{=}x_t) = P(\Phi{=}\phi_k \mid X{=}x_t) \odot f_B(k, \omega t, \omega(T-t+1)) \tag{13}$$

for $k = \{0, \ldots, N\}$, where α, β are hyperparameters of beta function $B(\cdot, \cdot)$, N is number of tokens and ω is scaling factor controlling width of prior: lower the ω, wider the width.

3. EXPERIMENTS

We evaluate the effectiveness of the alignment learning framework by comparing its performance in terms of convergence speed, distance from human annotated ground truth durations, and speech quality. We use the LJ Speech dataset[3] (LJ) [18] for all our experiments. For autoregressive models, we compare with the baseline alignment methods therein. For parallel models, we compare with alignment method that relies on an external TTS model (Tacotron2) or external aligner (MFA).

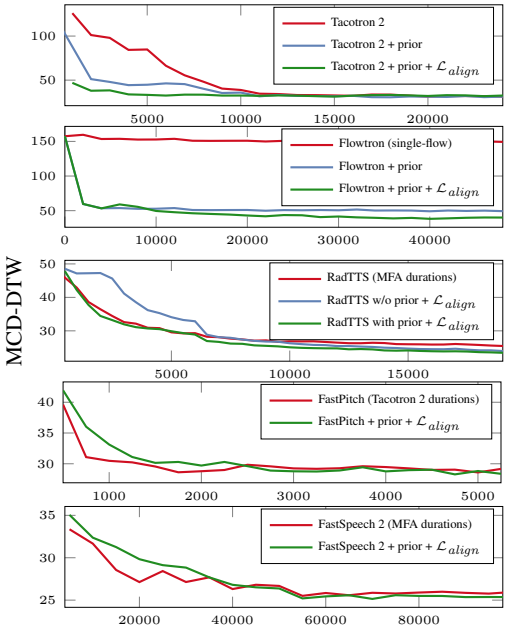
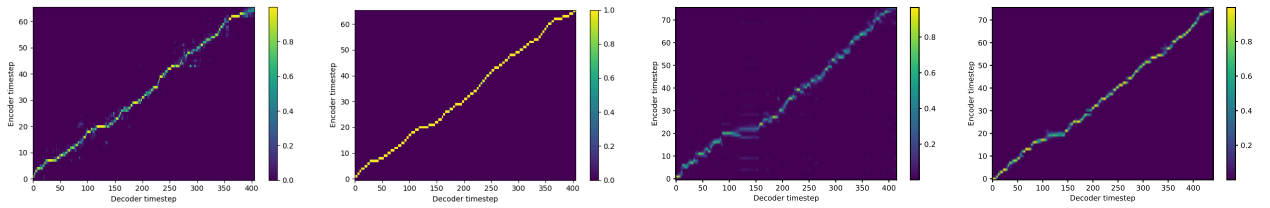


Fig. 2: MCD-DTW convergence rate improvements in TTS models with the alignment learning framework.

3.1. Convergence Rate

We use the mean mel-cepstral distance(MCD) [17, 19] to compare convergence rates. MCD compares the distance between synthesized and ground truth mel-spectrograms aligned temporally with dynamic time warping. We observe in Figure 2 that using the static prior described in Section 2.4 significantly improves the convergence rate of Tacotron2. Parallel models such as RAD-TTS, FastPitch and FastSpeech2 with the alignment framework (no dependency on external aligners) converge at the same rate as their baseline models using a forced aligner. Flowtron benefits the most from using the alignment framework. It has two autoregressive flows running in opposing directions, each with its own learned alignment. If the alignment in the first flow fails, the second flow, which depends on the output of the first, will fail as well. Thus, the baseline flowtron training is time consuming as it must train each flow step in succession. By using just the attention prior, we are now able to train at least two flows simultaneously, with

[3]We demonstrate the aligner works in multi-speaker setting (LibriTTS with 247 speakers). Experiments & samples available on our website.



(a) Flowtron (b) Flowtron with $\mathcal{L}_{align}$ (c) Tacotron 2 (d) Tacotron 2 with $\mathcal{L}_{align}$

Fig. 3: Converged soft alignments for Flowtron, Tacotron2. Alignment framework provides sharper, more connected alignments.

further improvements in convergence speed with the addition of alignment learning $\mathcal{L}_{align}$ objective (described in Sec 2.1).

3.2. Alignment Sharpness

We visually inspect alignment matrices for a random validation sample in Figure 3. The objective $\mathcal{L}_{align}$ consistently improves the sharpness and connectedness of the paths, leading to continuous speech without repeating or missing words.

3.3. Duration Analysis

To observe the influence of the alignment loss on the quality of alignments, we compare phoneme durations extracted from model alignments to manually annotated durations (due to lack of ground truth (GT) durations) from 10 samples of the LJ test set. For autoregressive models, we extract binarized alignments from soft alignments using monotonic argmax, iterating through phonemes and identifying the phoneme with maximum attention weights among the current and next phonemes. We use this binarized alignment to extract durations for each phoneme. Figure 4 shows average L_1 distance between durations extracted from the models with respect to ground truth annotated durations. The proposed framework leads to faster convergence rate and alignments that are closer to GT.

3.4. Pairwise Opinion Scores

We crowd-sourced pairwise preference scores to subjectively compare models. Listeners were pre-screened with a hearing test based on sinusoid counting. To perform pairwise ranking, raters were repeatedly given two synthesized utterances of the same text, picked at random from 100 LJ test samples. Both were synthesized with the same architecture: one being the baseline, and other with alignment framework. The listeners were shown the text and asked to select samples with best overall quality, defined by accuracy of text, pleasantness, and naturalness. Approximately 200 scores per model were collected. Table 1 shows pairwise preference scores of models trained with alignment framework over baseline. It shows that alignment framework consistently improves over all baselines.

Table 1: Pairwise preference scores by human raters, shown with 95% confidence intervals. Scores above 0.5 indicate models trained with $\mathcal{L}_{align}$ were preferred by majority of raters.

Model	Alignment Framework vs Baseline
Tacotron 2	0.556 ± 0.068
Flowtron ($\sigma = .5$)	0.635 ± 0.065
RAD-TTS ($\sigma = .5$)	0.639 ± 0.066
FastPitch	0.565 ± 0.068
FastSpeech2	0.521 ± 0.067

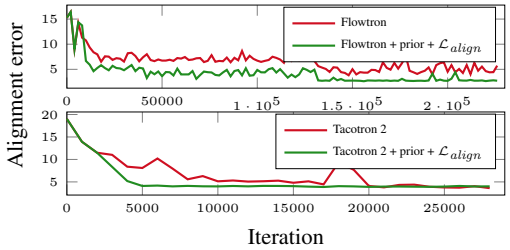


Fig. 4: L_1 distance between ground truth alignments and those extracted during training for Flowtron and Tacotron 2. Both use different batch sizes and are thus plotted separately.

3.5. Robustness to Errors on Long Utterances

We measure character error rate (CER) between synthesized and input texts using an external speech recognition model to evaluate the robustness of the alignments on long utterances. We use $14,045$ full sentences from the LibriTTS dataset [20]. We synthesize speech with models trained on LJ Speech, and transcribe it with Jasper [21]. Figure 5 shows that autoregressive models with $\mathcal{L}_{align}$ have a lower CER, providing evidence that the alignment objective results in more robust speech for long utterances. Parallel models such as RAD-TTS use a duration predictor and do not suffer from alignment issues, and hence have a much lower CER than autoregressive models.

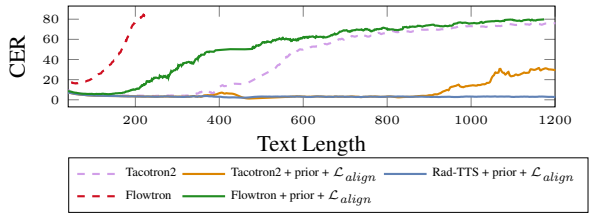


Fig. 5: Character error rate of models at different text lengths. Alignment framework results in fewer errors on long prompts.

4. CONCLUSION

We present an alignment framework that is broadly applicable to various TTS architectures, both autoregressive and parallel. This framework combines the forward-sum, Viterbi algorithm and a simple prior to make attention-based online alignment learning stable and fast-converging. It eliminates the need for forced aligners which are expensive to use and often not readily available for certain languages. Our experiments demonstrate improvements in speech quality based on human pairwise comparisons, reduced alignment failures, faster convergence, and robustness to errors in synthesis of long text sequences.

5. REFERENCES

[1] Yuxuan Wang, R. J. Skerry-Ryan, Daisy Stanton, Yonghui Wu, Ron J. Weiss, Navdeep Jaitly, Zongheng Yang, Ying Xiao, Zhifeng Chen, Samy Bengio, Quoc V. Le, Yannis Agiomyrgiannakis, Rob Clark, and Rif A. Saurous, "Tacotron: A fully end-to-end text-to-speech synthesis model," *CoRR*, vol. abs/1703.10135, 2017.

[2] Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, R. J. Skerry-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, and Yonghui Wu, "Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions," *CoRR*, vol. abs/1712.05884, 2017.

[3] Rafael Valle, Kevin J. Shih, Ryan Prenger, and Bryan Catanzaro, "Flowtron: an autoregressive flow-based generative network for text-to-speech synthesis," in *International Conference on Learning Representations*, 2021.

[4] Yi Ren, Chenxu Hu, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu, "Fastspeech 2: Fast and high-quality end-to-end text-to-speech," *arXiv preprint arXiv:2006.04558*, 2020.

[5] Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu, "Fastspeech: Fast, robust and controllable text to speech," in *Advances in Neural Information Processing Systems*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. 2019, vol. 32, pp. 3171–3180, Curran Associates, Inc.

[6] Jaehyeon Kim, Sungwon Kim, Jungil Kong, and Sungroh Yoon, "Glow-tts: A generative flow for text-to-speech via monotonic alignment search," *Advances in Neural Information Processing Systems*, vol. 33, pp. 8067–8077, 2020.

[7] Adrian Łańcucki, "Fastpitch: Parallel text-to-speech with pitch prediction," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6588–6592.

[8] Alex Graves, "Generating sequences with recurrent neural networks," *CoRR*, vol. abs/1308.0850, 2013.

[9] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[10] Michael McAuliffe, Michaela Socolof, Sarah Mihuc, M. Wagner, and Morgan Sonderegger, "Montreal forced aligner: Trainable text-speech alignment using kaldi," in *INTERSPEECH*, 2017.

[11] Kainan Peng, Wei Ping, Zhao Song, and Kexin Zhao, "Non-autoregressive neural text-to-speech," in *International Conference on Machine Learning*. PMLR, 2020, pp. 7586–7598.

[12] Kevin J. Shih, Rafael Valle, Rohan Badlani, Adrian Lancucki, Wei Ping, and Bryan Catanzaro, "RAD-TTS: Parallel flow-based TTS with robust alignment learning and diverse synthesis," in *ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models*, 2021.

[13] Hideyuki Tachibana, Katsuya Uenoyama, and Shunsuke Aihara, "Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention," *CoRR*, vol. abs/1710.08969, 2017.

[14] Lawrence R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[15] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," New York, NY, USA, 2006, ICML '06, p. 369–376, Association for Computing Machinery.

[16] Jan Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, KyungHyun Cho, and Yoshua Bengio, "Attention-based models for speech recognition," *CoRR*, vol. abs/1506.07503, 2015.

[17] E. Battenberg, R. J. Skerry-Ryan, S. Mariooryad, D. Stanton, D. Kao, M. Shannon, and T. Bagby, "Location-relative attention mechanisms for robust long-form speech synthesis," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 6194–6198.

[18] Keith Ito and Linda Johnson, "The lj speech dataset," `https://keithito.com/LJ-Speech-Dataset/`, 2017.

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

[20] Heiga Zen, Rob Clark, Ron J. Weiss, Viet Dang, Ye Jia, Yonghui Wu, Yu Zhang, and Zhifeng Chen, "Libritts: A corpus derived from librispeech for text-to-speech," in *Interspeech*, 2019.

[21] Jason Li, Vitaly Lavrukhin, Boris Ginsburg, Ryan Leary, Oleksii Kuchaiev, Jonathan Cohen, Huyen Nguyen, and Ravi Gadde, "Jasper: An end-to-end convolutional neural acoustic model," Interspeech 2019.