

# Deep Learning for Audio

## Lecture 5

Pavel Severilov

AI Masters

2023

# Outline

1. Text-to-speech
2. Attention ideas in TTS
3. SOTA TTS models

# Outline

1. Text-to-speech
2. Attention ideas in TTS
3. SOTA TTS models

# Text-to-speech (TTS)

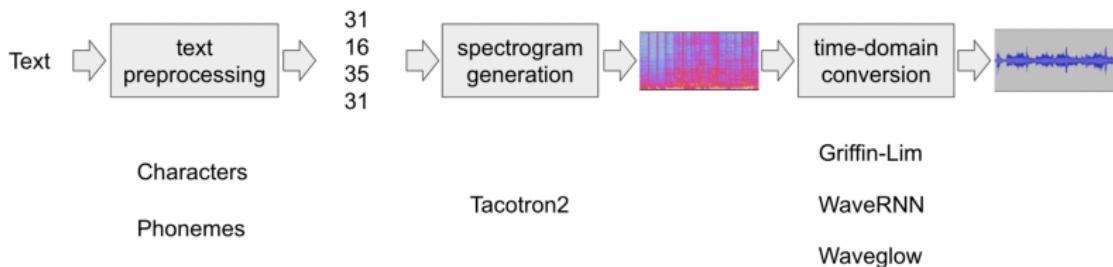


Figure: Base pipeline of Text-to-speech task

Datasets: LJSpeech, LibriTTS, CommonVoice, OpenTTS (Ru)

## TTS: quality

- ▶ Quality: Subjective perception
- ▶ Overall impression, Intelligibility, Similarity, Naturalness, Pleasantness, Intonation and pauses, Emotions, Listening effort
- ▶ Mean Opinion Score (MOS): Crowdsourcing by Yandex Toloka/Amazon

$$\text{MOS} = \frac{\sum_{n=1}^N \mathcal{R}_n}{N}$$

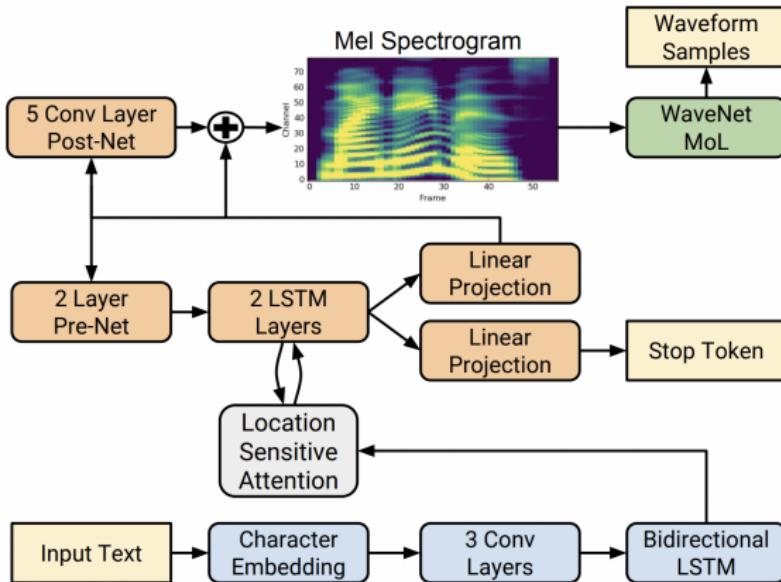
Rating	Quality	Distortion
5	Excellent	Imperceptible
4	Good	Just perceptible, but not annoying
3	Fair	Perceptible and slightly annoying
2	Poor	Annoying, but not objectionable
1	Bad	Very annoying and objectionable

- ▶ Side-by-Side audio comparison (evaluate small improvements)

# Outline

1. Text-to-speech
2. Attention ideas in TTS
3. SOTA TTS models

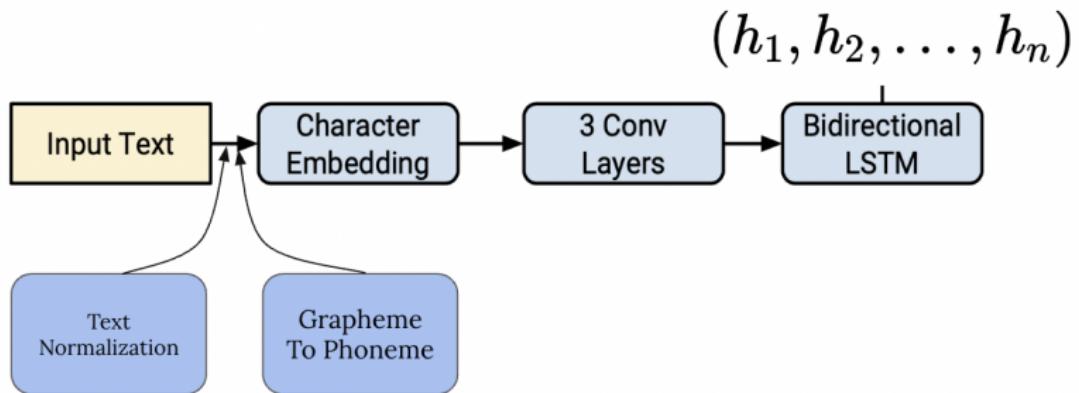
# Tacotron 2



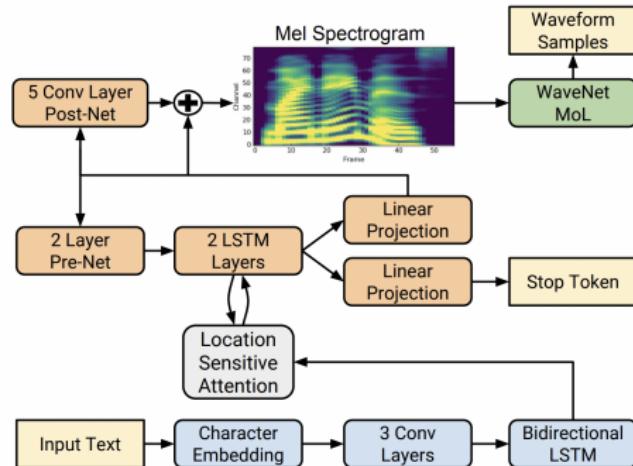
**Figure:** Recurrent seq-to-seq feature prediction network with attention that maps character embeddings to spectrograms, followed by a modified WaveNet to synthesize time-domain waveforms

*Shen et al., Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions, 2018 IEEE ICASSP*

## Tacotron 2: text Encoder



# Tacotron 2



$$\mathcal{L} = \mathcal{L}_{\text{pre}} + \mathcal{L}_{\text{post}} + \text{StopToken}$$

$$\mathcal{L}_{\text{pre}} = \text{MSE}(x, \hat{x}_{\text{pre}})$$

$$\mathcal{L}_{\text{post}} = \text{MSE}(x, \hat{x}_{\text{post}})$$

$$\text{StopToken} = \text{CE}(a, \mathcal{I}[h = \text{Stop}])$$

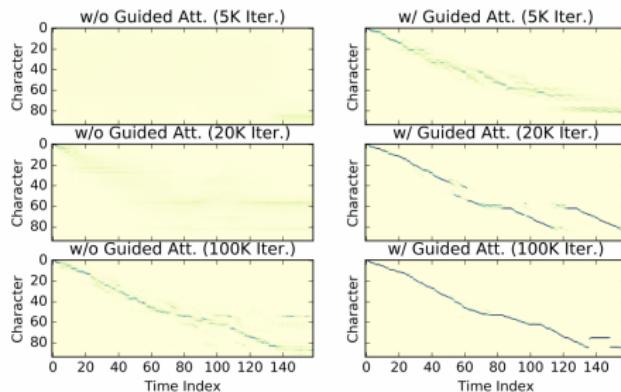
# Guided Attention

- Idea: text position  $n$  progresses nearly linearly to the time  $t$ :

$$n \sim at, \quad a \sim N/T$$

- Add loss  $\mathcal{L}_{\text{att}}(A) = \mathbb{E}_{nt}[A_{nt} W_{nt}]$ , where

$$W_{nt} = 1 - \exp\{-(n/N - t/T)^2/2g^2\}, \quad g = 0.2, \quad A \in \mathbb{R}^{N \times T}$$



**Figure:** Comparison of the attention matrix  $A$ , trained with (Right) and without (Left) the guided attention loss

# Monotonic Attention

$$e_{i,j} = a(s_{i-1}, h_j)$$

For  $j = t_{i-1}, t_{i-1} + 1, t_{i-1} + 2, \dots$ :  $p_{i,j} = \sigma(e_{i,j})$

$$z_{i,j} \sim \text{Bernoulli}(p_{i,j})$$

where  $a(\cdot)$  – learnable deterministic "energy function",  $\sigma(\cdot)$  – logistic sigmoid function

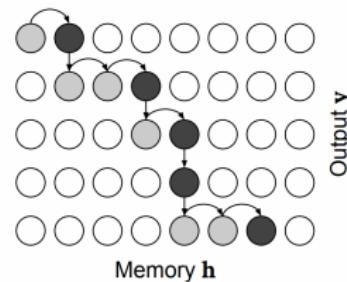
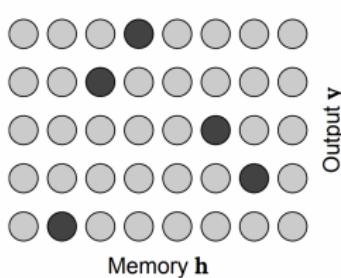
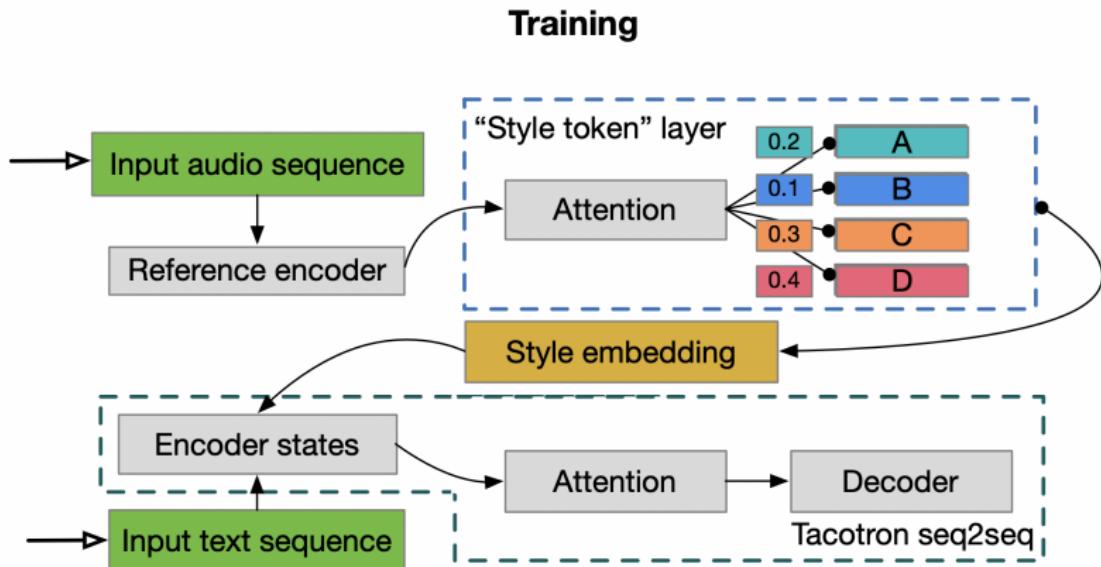


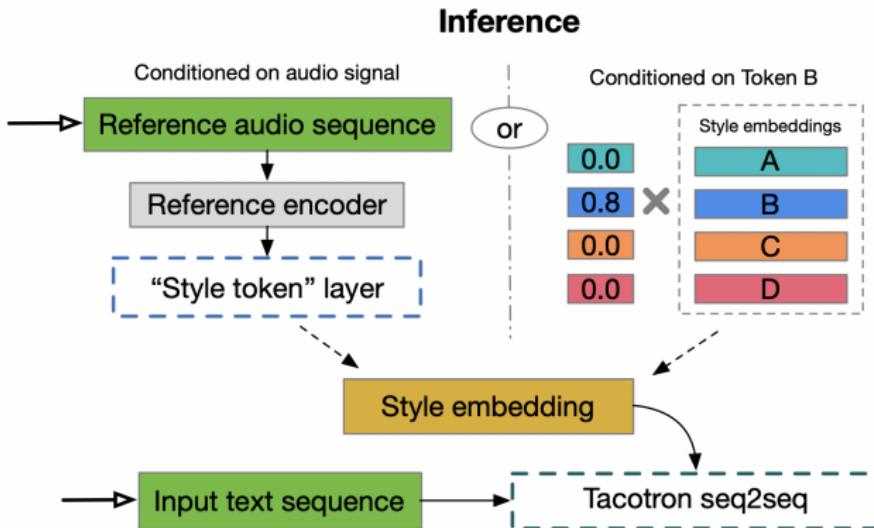
Figure: Left: schematic of the stochastic process underlying softmax-based attention decoders. Right: monotonic stochastic decoding process.

# Global Style Token (GST)



**Figure:** Log-mel spectrogram of the training target is fed to the reference encoder followed by a style token layer. The resulting style embedding is used to condition the Tacotron text encoder states

# Global Style Token (GST)

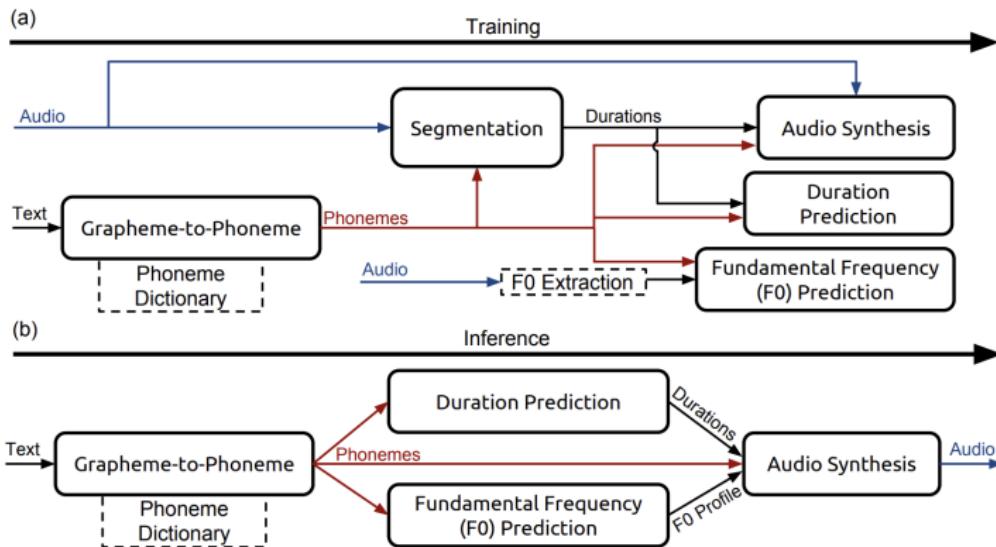


**Figure:** Inference: can feed an arbitrary reference signal to synthesize text with its speaking style. Alternatively, can remove the reference encoder and directly control synthesis using the learned interpretable tokens.

# Outline

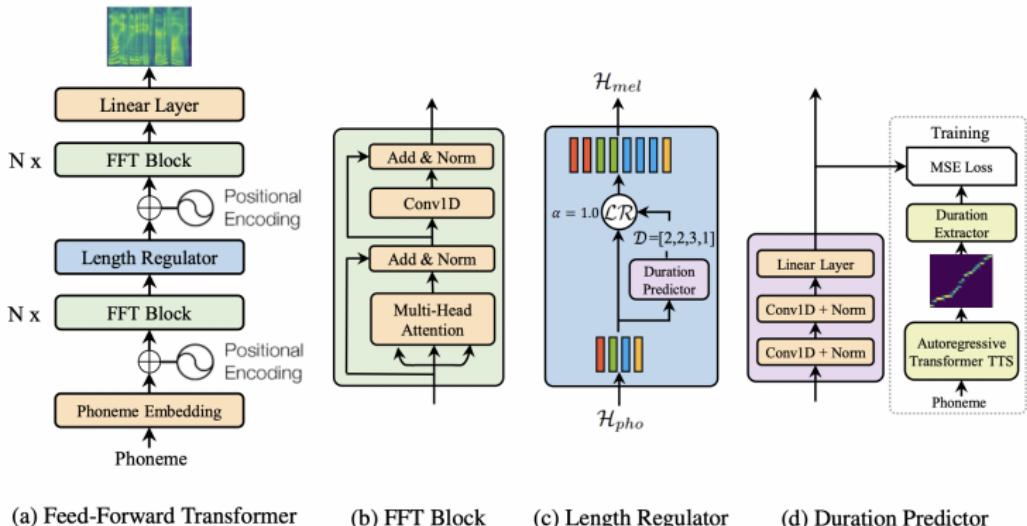
1. Text-to-speech
2. Attention ideas in TTS
3. SOTA TTS models

# DeepVoice



- ▶ Grapheme-to-phoneme model – for words not present in a phoneme dictionary
- ▶ Segmentation model identifies where in the audio each phoneme begins and ends

# FastSpeech

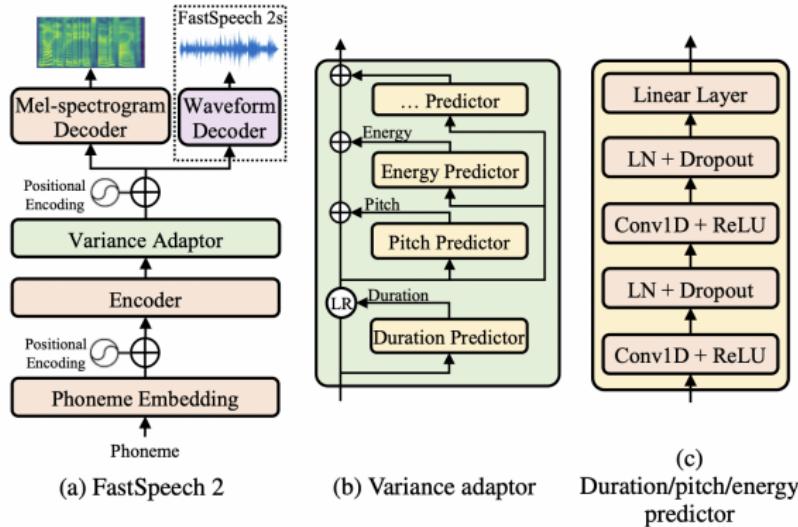


- ▶ Use of feed-forward Transformer (FFT) blocks
- ▶ Extract attention alignments from an encoder-decoder based teacher model for phoneme duration prediction

## FastSpeech vs Tacotron

	Tacotron	Fastspeech
<b>Inference speed</b>	Slow	Fast
<b>Synthesized speech is robust?</b>	No, some words are skipped or repeated	Yes
<b>Controllability (voice speed or prosody control)</b>	Lack	Adjust voice speed smoothly

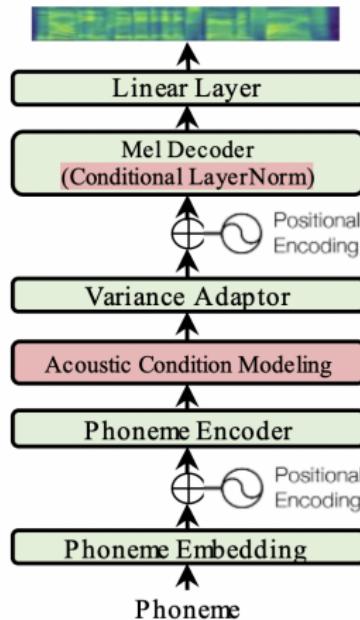
# FastSpeech 2



- ▶ Directly training the model with ground-truth target instead of the simplified output from teacher
- ▶ more information of speech used as conditional inputs:  
duration, pitch (emotions), energy (volume and prosody)

*Ren, Yi et al. FastSpeech 2: Fast and High-Quality End-to-End Text to Speech, Arxiv preprint, 2021*

# AdaSpeech



- ▶ Adopted FastSpeech 2
- ▶ Add acoustic condition modeling

Chen et al., AdaSpeech: Adaptive Text to Speech for Custom Voice, Arxiv preprint, 2021

# AdaSpeech: acoustic condition modeling

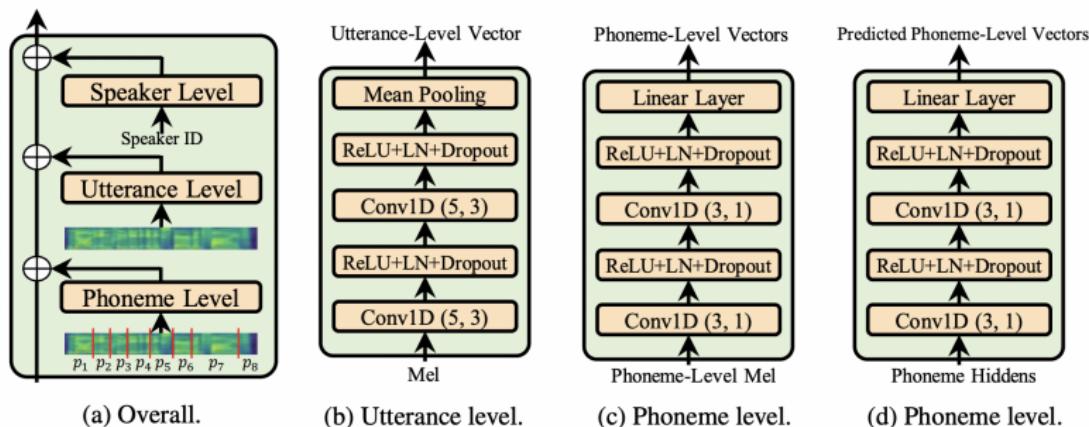


Figure: Adding acoustic conditions such as speaker timbre, prosody and recording environments into model