The 12th ISCSLP 2021 第12届
International Symposium on Chinese Spoken Language Processing
中文口語語言處理國際會議

Microsoft

Pushing the Frontier of Neural Text to Speech

Xu Tan, Senior Researcher
Microsoft Research Asia

xuta@microsoft.com

Self-introduction

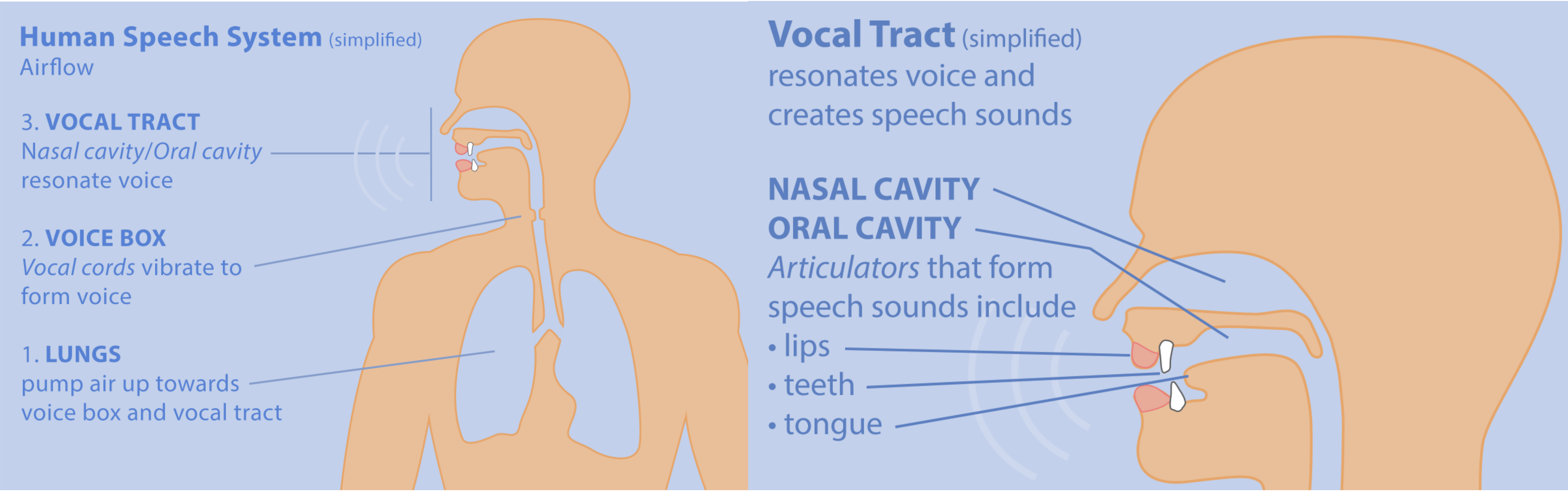
- Xu Tan (谭旭)
- Senior Researcher @ Machine Learning Group, Microsoft Research Asia
- Research interests: deep learning and its applications on NLP and Speech
  - Text to speech
  - Automatic speech recognition
  - Neural machine translation
  - Language/speech pre-training
  - Music understanding and generation
- Homepage: https://www.microsoft.com/en-us/research/people/xuta/
- Speech related research: https://speechresearch.github.io/

Outline

- Overview of text to speech
- Pushing the frontier of neural text to speech
  - More end-to-end
  - Inference speedup
  - Robustness, expressiveness and controllability
  - Low-resource
  - From research to product
- Summary

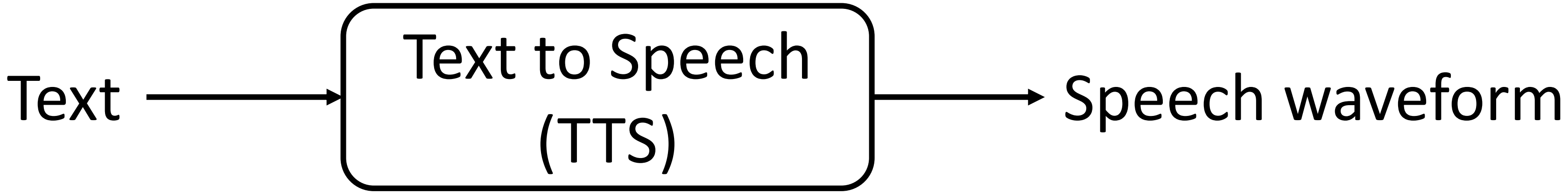
Text to speech synthesis

- The artificial production of human speech from text
  - Human speech system



Text to speech synthesis

- The artificial production of human speech from text



- Disciplines: acoustics, linguistics, digital signal processing, statistics and deep learning
- The quality of the synthesized speech is measured by
  - Intelligibility and naturalness
  - From intelligibility to naturalness

History of TTS Technology

- Concatenative speech synthesis
  - High intelligibility, but requires huge database, less natural and emotionless
- Statistical parametric speech synthesis
  - Lower data cost and more flexible, but lower quality and robotic
- Neural network based end-to-end speech synthesis
  - Huge quality improvement, less human preprocessing and feature development

Concatenative

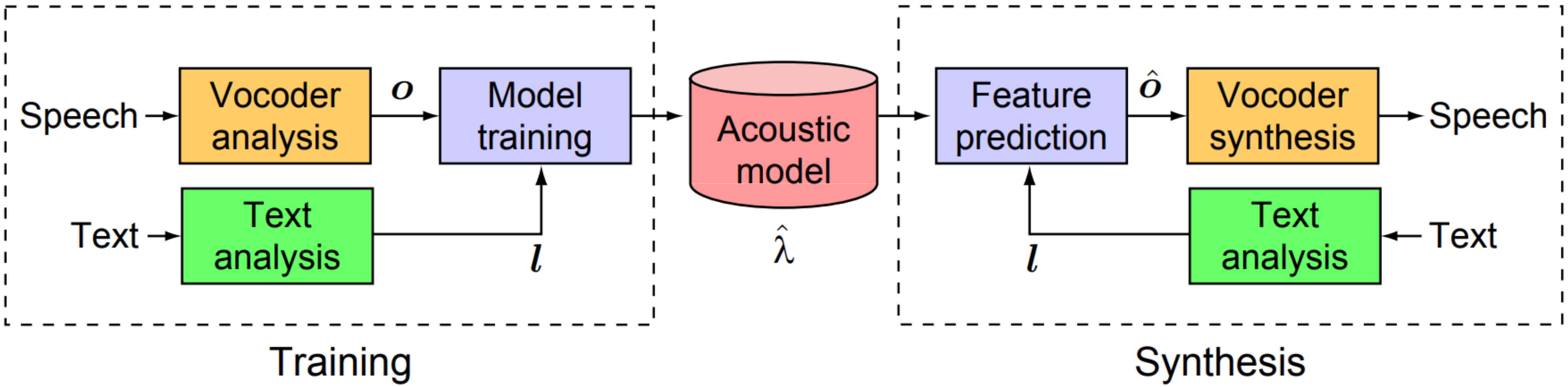
Statistical parametric (HMM)

Neural (Tacotron 2)

Neural (FastSpeech 2)

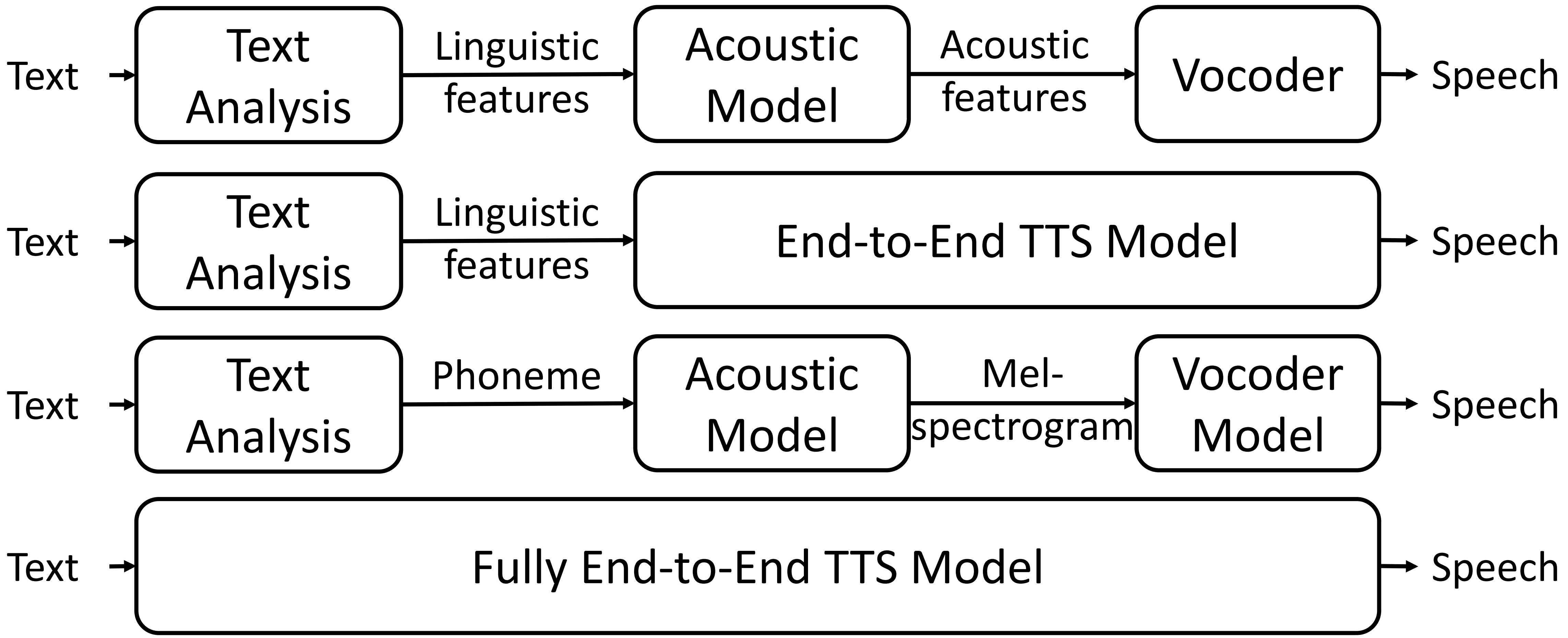
Statistical parametric speech synthesis

- Text analysis, acoustic model, and vocoder analysis/synthesis



- Text analysis: text → linguistic features
- Acoustic model: linguistic features → acoustic features
- Vocoder analysis: speech → acoustic features
- Vocoder synthesis: acoustic features → speech

Neural based end-to-end speech synthesis



Text analysis

- Transforms input text into linguistic features, including
  - Text normalization
    - 1989 → nineteen eighty nine, *Jan. 24th* → *January twenty-fourth*
  - Phrase/word/syllable segmentation
    - synthesis → syn-the-sis
  - Part of speech (POS) tagging
    - Mary went to the store → noun, verb, prep, noun,
  - ToBI (Tones and Break Indices)
    - Mary went to the store ? → Mary' store' H%
  - Grapheme-to-phoneme conversion
    - *Speech* → *s p iy ch*

Text analysis——Linguistic features

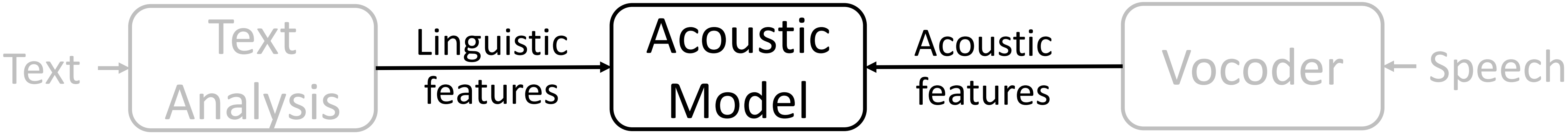
- Phoneme, syllable, word, phrase and sentence-level features, e.g.,
  - The phonetic symbols of the previous before the previous, the previous, the current, the next or the next after the next;
  - Whether the previous, the current or the next syllable is stressed;
  - The part of speech (POS) of the previous, the current or the next word;
  - The prosodic annotation of the current phrase;
  - The number of syllables, words or phrases in the current sentence.

Text analysis——Linguistic features

- phoneme:
  - current phoneme
  - preceding and succeeding two phonemes
  - position of current phoneme within current syllable
- syllable:
  - numbers of phonemes within preceding, current, and succeeding syllables
  - stress[3] and accent[4] of preceding, current, and succeeding syllables
  - positions of current syllable within current word and phrase
  - numbers of preceding and succeeding stressed syllables within current phrase
  - numbers of preceding and succeeding accented syllables within current phrase
  - number of syllables from previous stressed syllable
  - number of syllables to next stressed syllable
  - number of syllables from previous accented syllable
  - number of syllables to next accented syllable
  - vowel identity within current syllable
- word:
  - guess at part of speech of preceding, current, and succeeding words
  - numbers of syllables within preceding, current, and succeeding words
  - position of current word within current phrase
  - numbers of preceding and succeeding content words within current phrase
  - number of words from previous content word
  - number of words to next content word
- phrase:
  - numbers of syllables within preceding, current, and succeeding phrases
  - position of current phrase in major phrases
  - ToBI endtone of current phrase
- utterance:
  - numbers of syllables, words, and phrases in utterance

Acoustic model

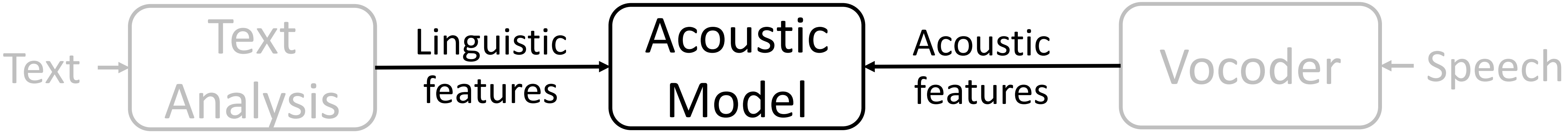
- Predict acoustic features from linguistic features



  - F0, V/UV, energy
  - Mel-scale Frequency Cepstral Coefficients (MFCC), Bark-Frequency Cepstral Coefficients (BFCC)
  - Mel-generalized coefficients (MGC), band aperiodicity (BAP),
  - Linear prediction coefficient (LPC),
  - Mel-spectrogram
    - Pre-emphasis, Framing, Windowing, Short-Time Fourier Transform (STFT), Mel filter

Acoustic model

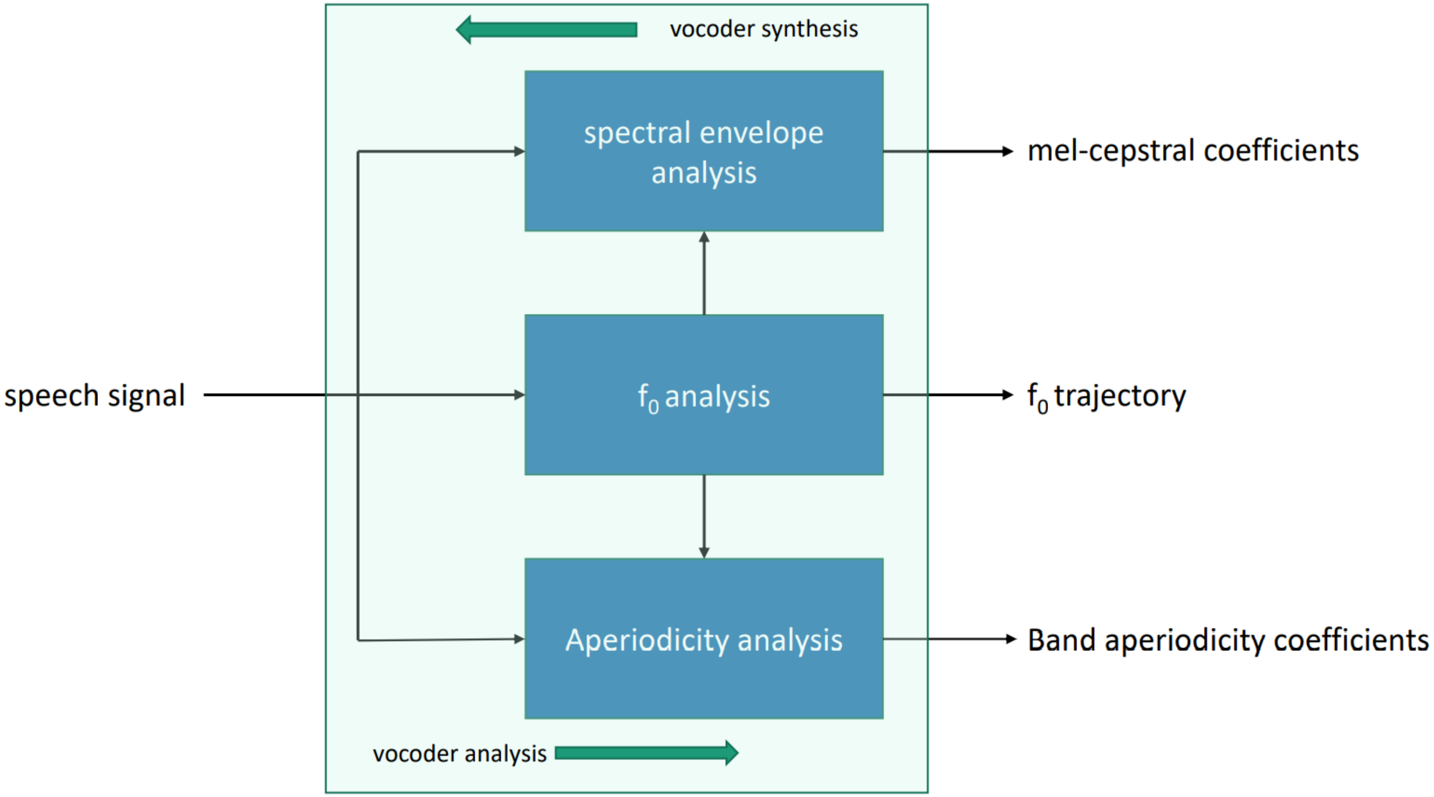
- Predict acoustic features from linguistic features



  - HMM, BLSTM, Seq2Seq (LSTM, CNN, Transformer)
  - The requirements for acoustic model
    - More context information (input)
    - Model correlation between frames (output)
    - Combat over-smoothing prediction
    - Alignment between linguistic and acoustic features

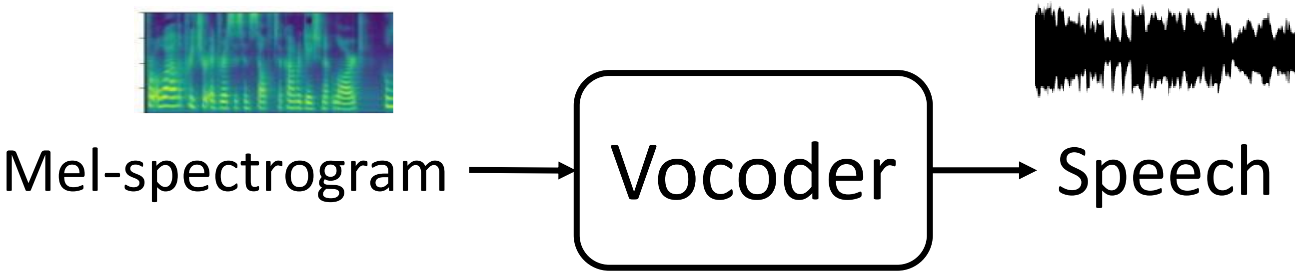
Vocoder

- Statistical parametric speech synthesis
  - HTS, STRAIGHT, Phase vocoder, PSOLA, sinusoidal model, WORLD



Vocoder

- Neural vocoder



  - WaveNet, ParallelWaveNet
  - SampleRNN, WaveRNN, LPCNet
  - GAN-based model
  - Flow-based model
  - Diffusion-based model

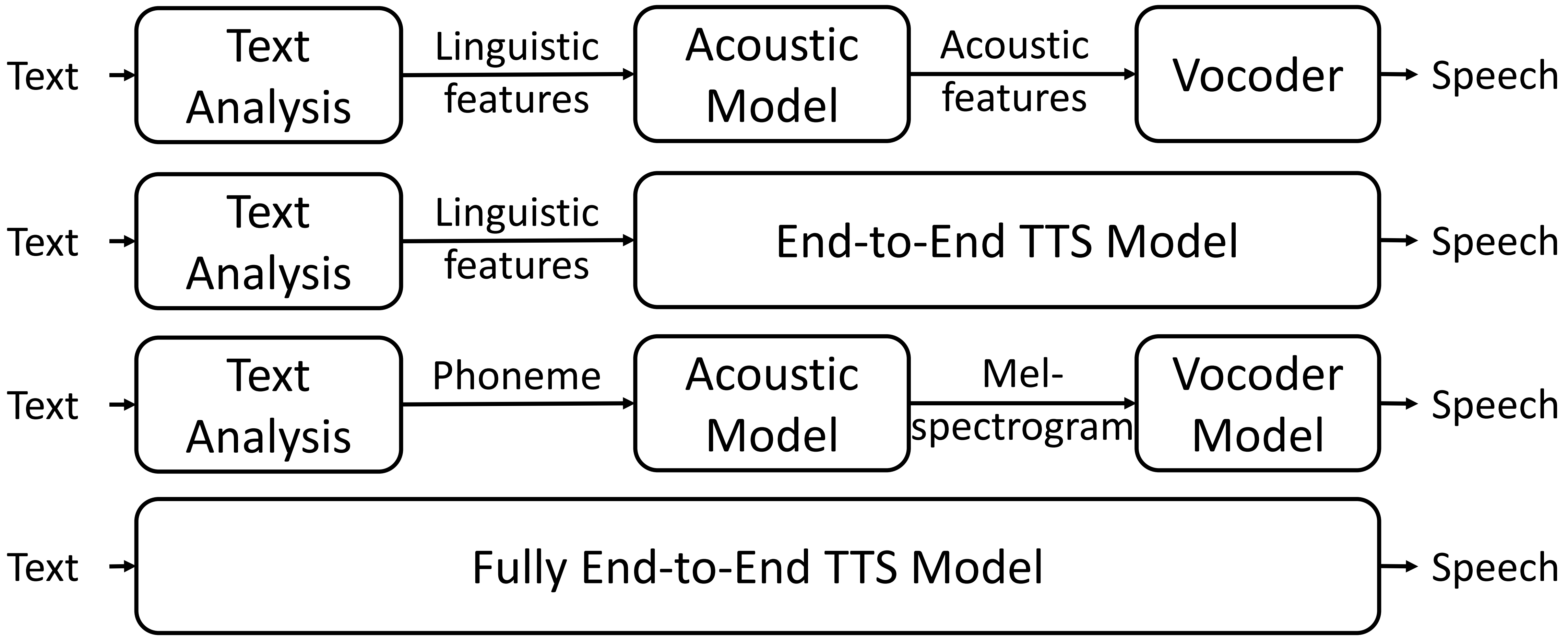
Outline

- Overview of text to speech
- Pushing the frontier of neural text to speech
  - More end-to-end
  - Inference speedup
  - Robustness, expressiveness and controllability
  - Low-resource
  - From research to product
- Summary

More end-to-end TTS

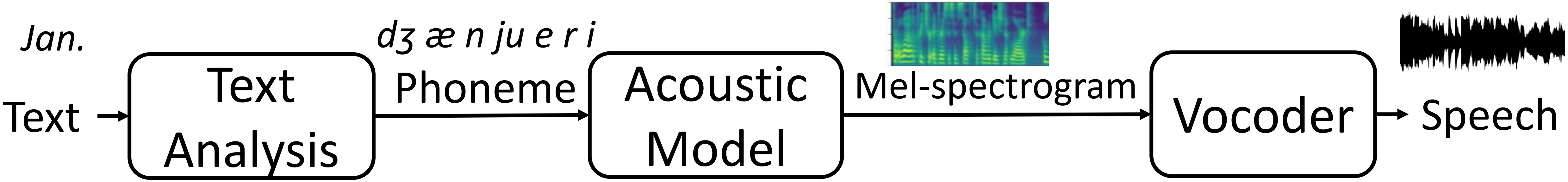
- Advantages of end-to-end model
  - Trained with text-speech pairs with minimum human annotation
  - Do not require explicit alignment between text and speech
  - Errors cannot accumulate and no error propagation since it is a single model
- Progressively end-to-end
  - WaveNet [6], DeepVoice [18], Tacotron [21], Char2Wav [23], DeepVoice 2 [19]
  - Tacotron 2 [22], DeepVoice 3 [20], Transformer TTS [25], FastSpeech [26]
  - ClariNet [24], EATS [28], FastSpeech 2s [27]

More end-to-end TTS



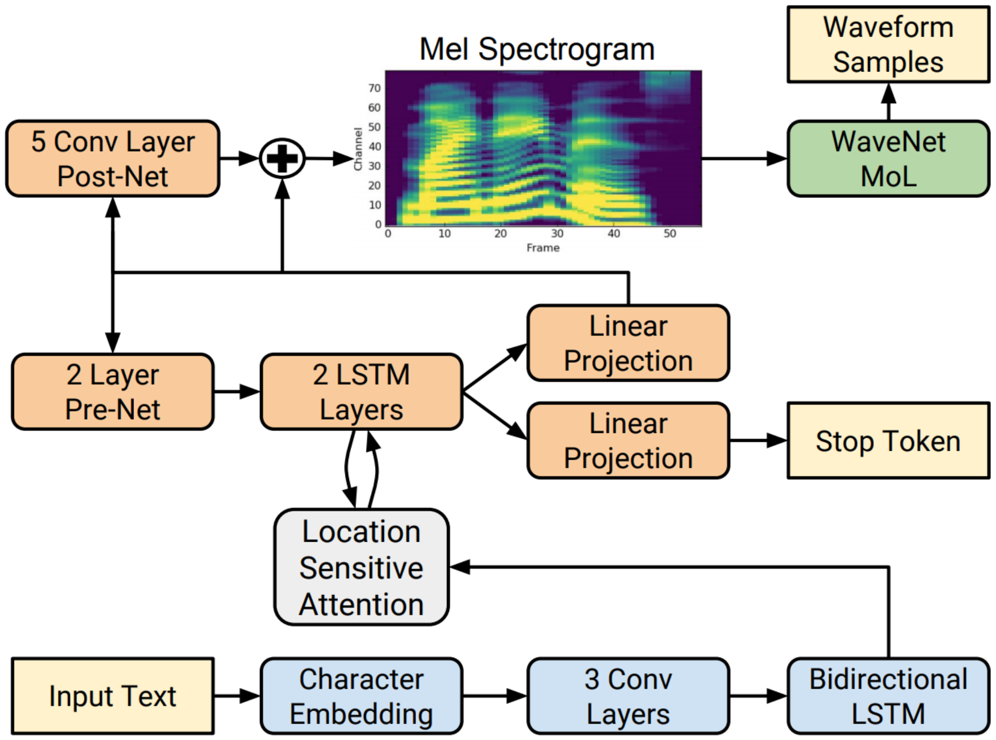
More end-to-end TTS

- Simplify/remove text analysis
  - Text normalization, phrase/word/syllable segmentation, POS tagging, ToBI, grapheme-to-phoneme conversion
  - Only text normalization and grapheme-to-phoneme conversion
    - *Jan. 24th → January twenty-fourth → dʒænjueri twenti fɔːrϑ*
- Simplify acoustic features
  - F0, MGC, BAP → mel-spectrogram



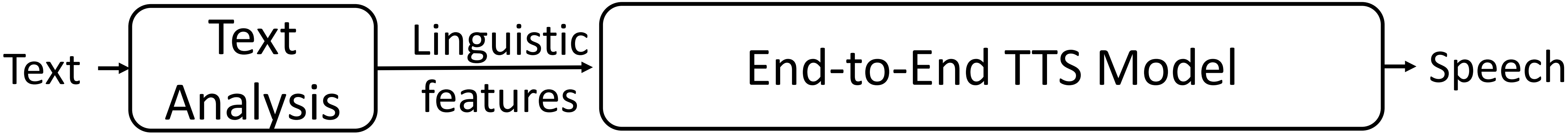
More end-to-end TTS

- Simplify/remove text analysis, and simplify acoustic features
  - Tacotron 2 [22]



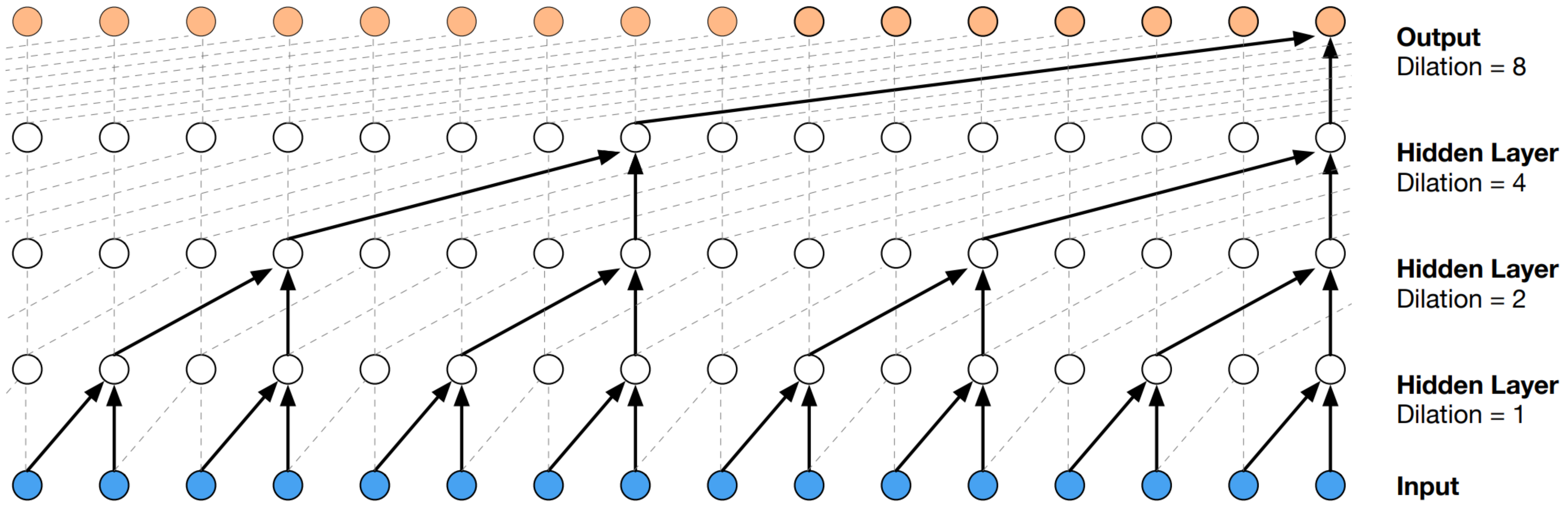
More end-to-end TTS

- Directly predict waveform instead of mel-spectrogram
  - WaveNet [6]: linguistic features, F0, duration → waveform



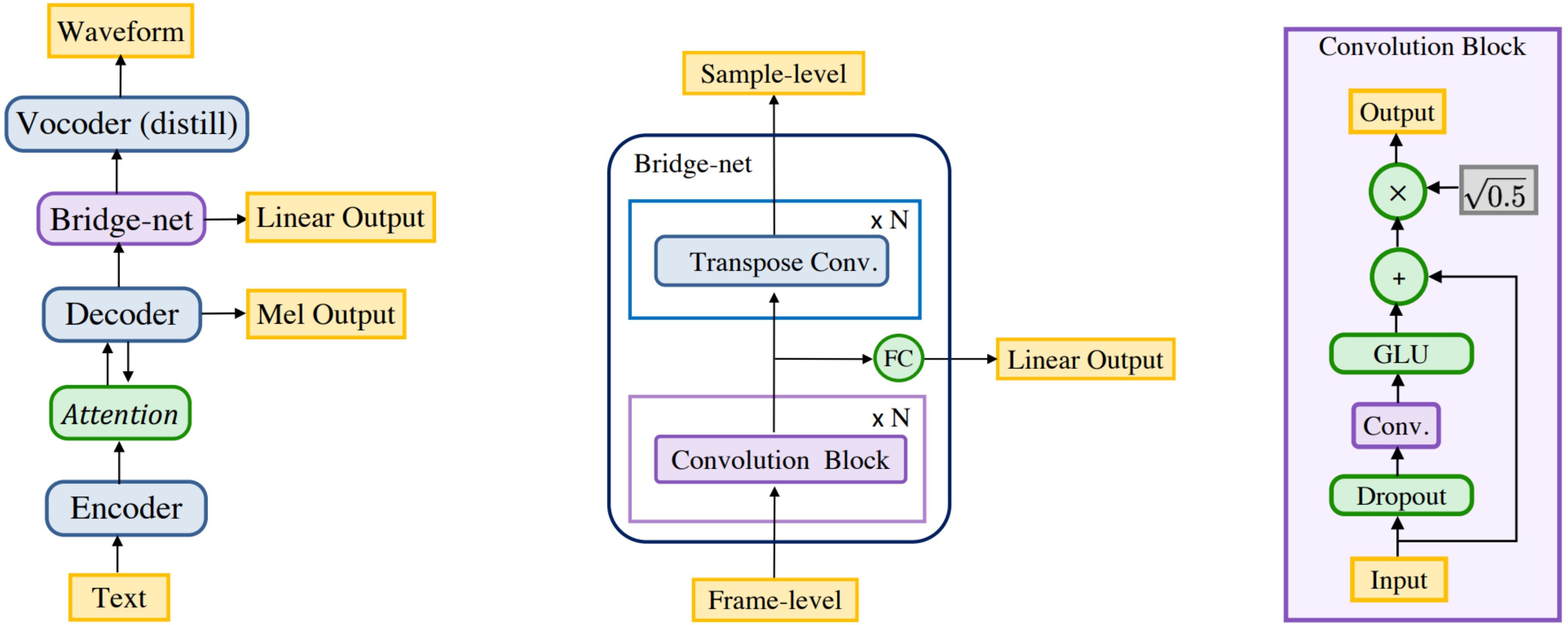
More end-to-end TTS

- Directly predict waveform instead of mel-spectrogram
  - WaveNet [6]: autoregressive model with dilated causal convolution



More end-to-end TTS

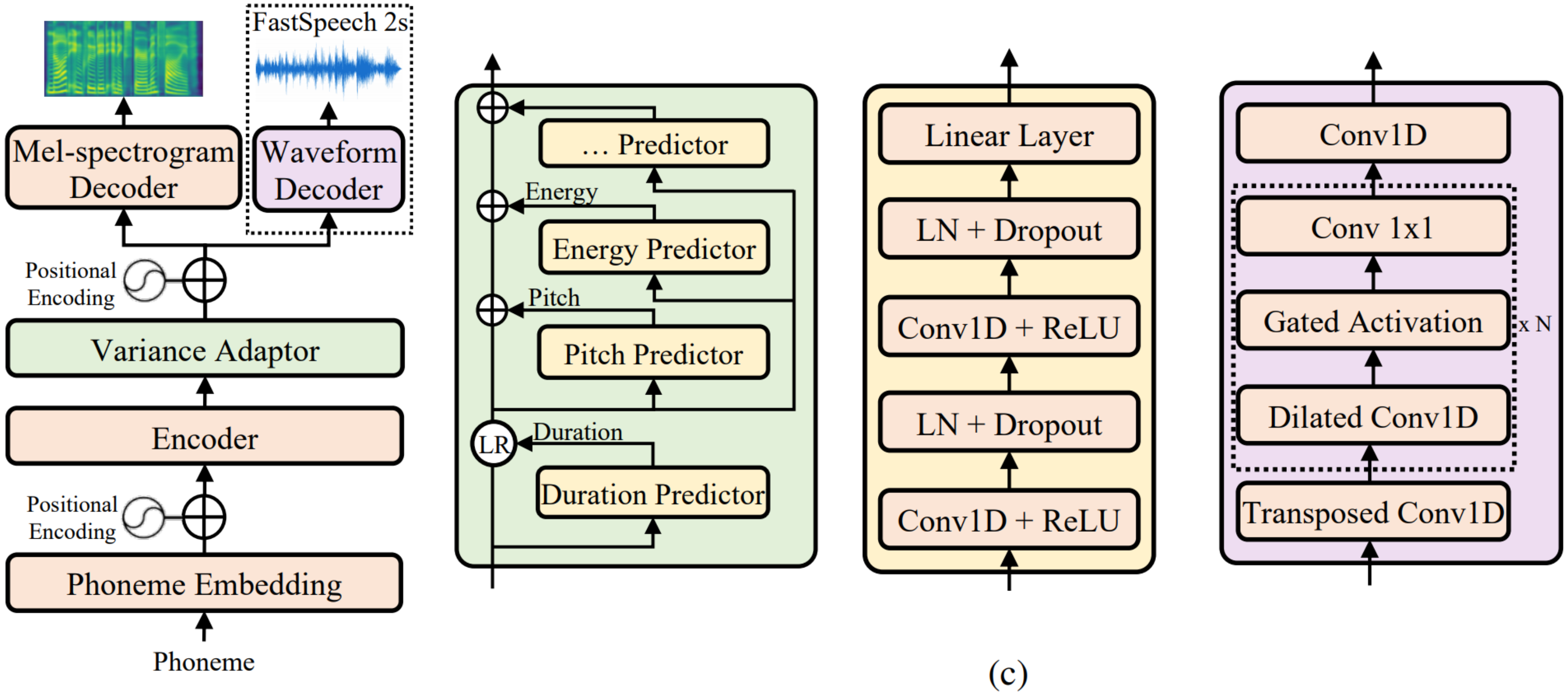
- Fully end-to-end, direct text to waveform synthesis
  - ClariNet [24]: autoregressive acoustic model and non-autoregressive vocoder



(a) Text-to-wave architecture (b) Bridge-net (c) Convolution block

More end-to-end TTS

- Fully end-to-end, direct text to waveform synthesis
  - FastSpeech 2s [27]: fully parallel text to wave model



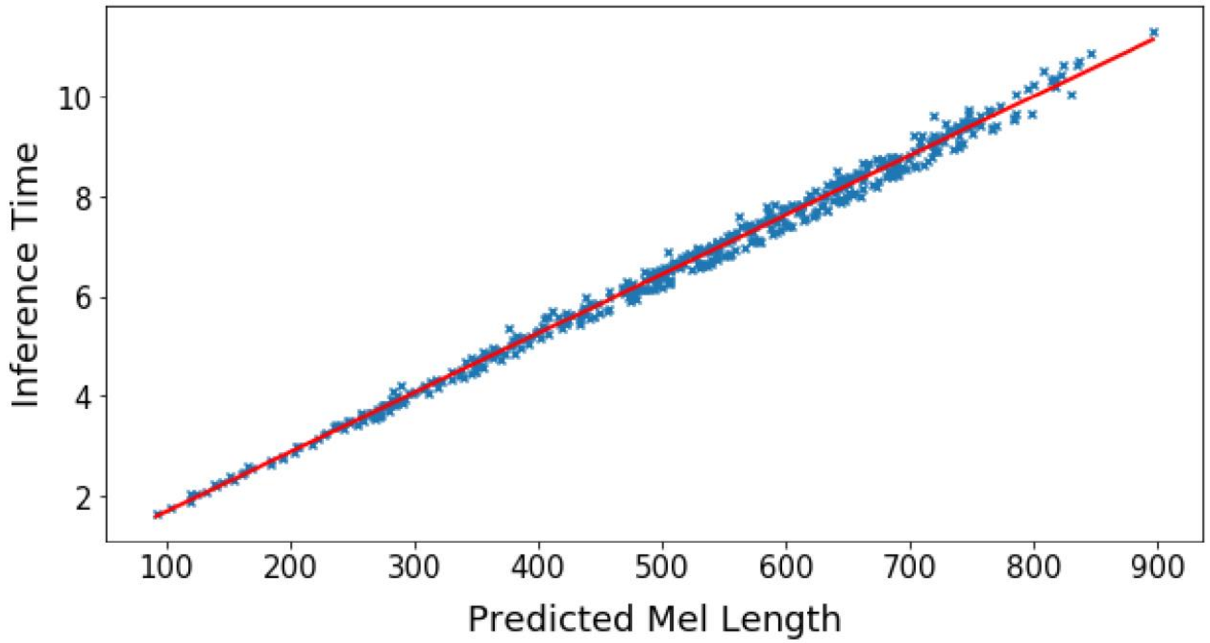
(a) FastSpeech 2 (b) Variance adaptor (c) Duration/pitch/energy predictor (d) Waveform decoder

Outline

- Overview of text to speech
- Pushing the frontier of neural text to speech
    - More end-to-end
    - Inference speedup
    - Robustness, expressiveness and controllability
    - Low-resource
    - From research to product
- Summary

Inference speedup

- End-to-end neural TTS model usually adopts autoregressive mel-spectrogram and waveform generation
  - Sequence is very long, e.g., 1s speech, 500 mel, 24000 waveform points
  - Slow inference speed



Inference speedup

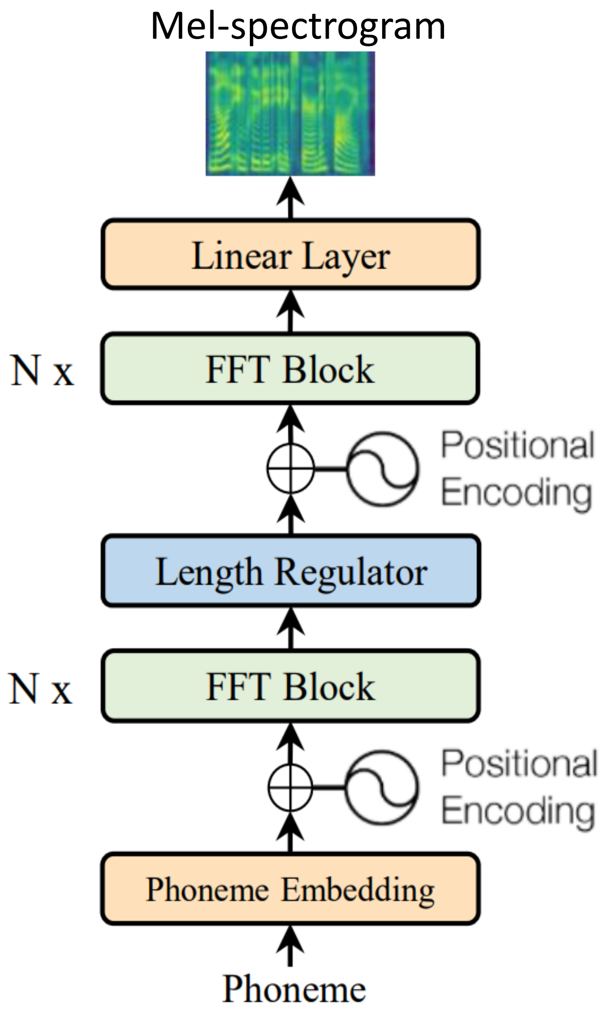
- Non-autoregressive mel-spectrogram generation
  - FastSpeech [26], FastSpeech 2 [27], ParaNet [29], Glow-TTS [30]
- Non-autoregressive vocoder
  - Parallel WaveNet [7]
  - GAN based: WaveGAN [14], MelGAN [15], Parallel WaveGAN [16], GAN-TTS [17], HiFi-GAN [36]
  - Flow based: WaveGlow [11], FloWaveNet [12], WaveFlow [13]
  - Diffusion-based: DiffWave [31], WaveGrad [32]
- Lightweight model
  - WaveRNN [9], LPCNet [10], multiband modeling [37,38], model compression [9]

Inference speedup——FastSpeech

- Problems: Previous autoregressive TTS models (Tacotron 2, DeepVoice 3, Transformer TTS) suffer from
  - Slow inference speed: autoregressive mel-spectrogram generation is slow for long sequence;
  - Not robust: words skipping and repeating;
  - Lack of controllability: hard to control the voice speed/prosody in the autoregressive generation

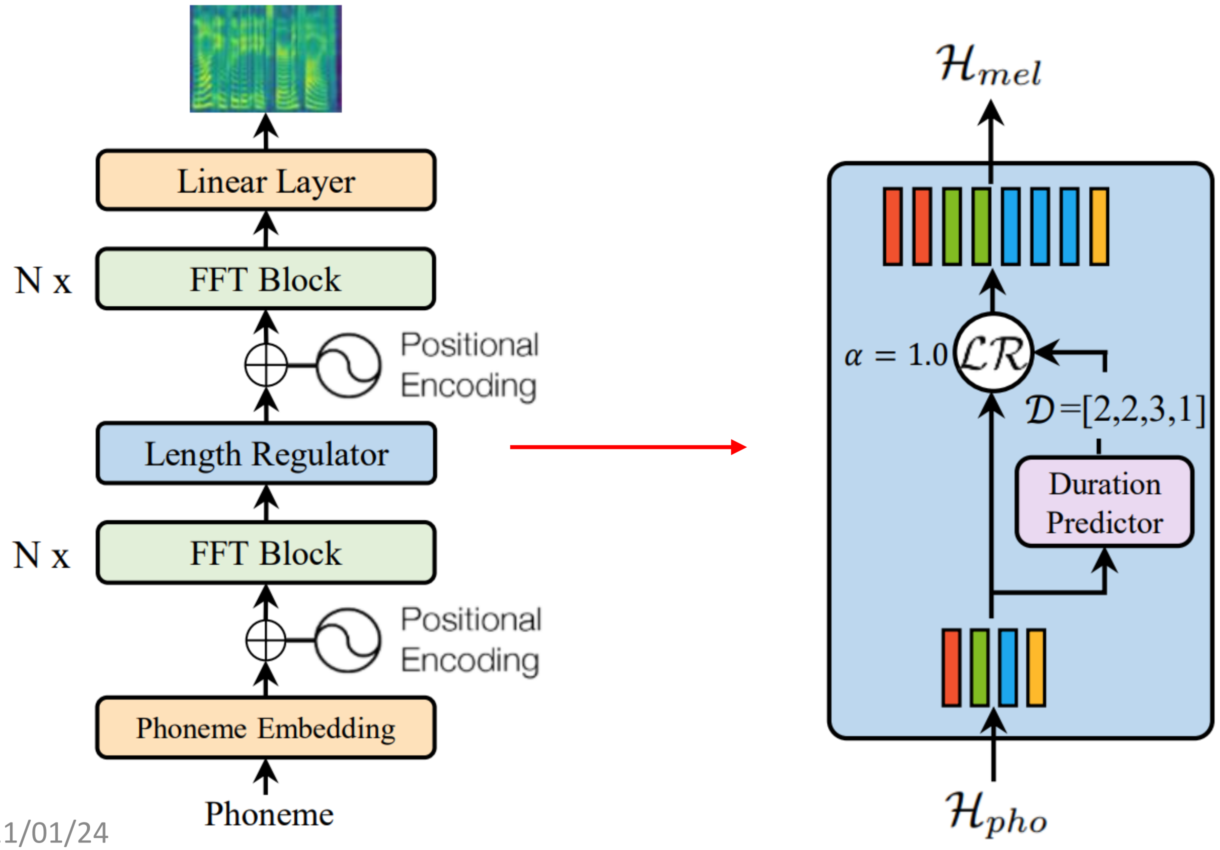
You can call me directly at 4257037344 or my cell 4254447474 or send me a meeting request with all the appropriate information.

- Key designs in FastSpeech [26]
  - Generate mel-spectrogram in parallel (for speedup)
  - Remove the text-speech attention mechanism (for robustness)
  - Feed-forward transformer with length regulator (for controllability)



Inference speedup——FastSpeech

- Framework: Length Regulator



speech

1.0x speed — s p iy ch ← 2 2 3 1 (α =1)

s s p p iy iy iy ch

2.0x speed — s p iy ch ← 2 2 3 1 (α = 0.5 (1 1 1.5 0.5 → 1 1 2 1))

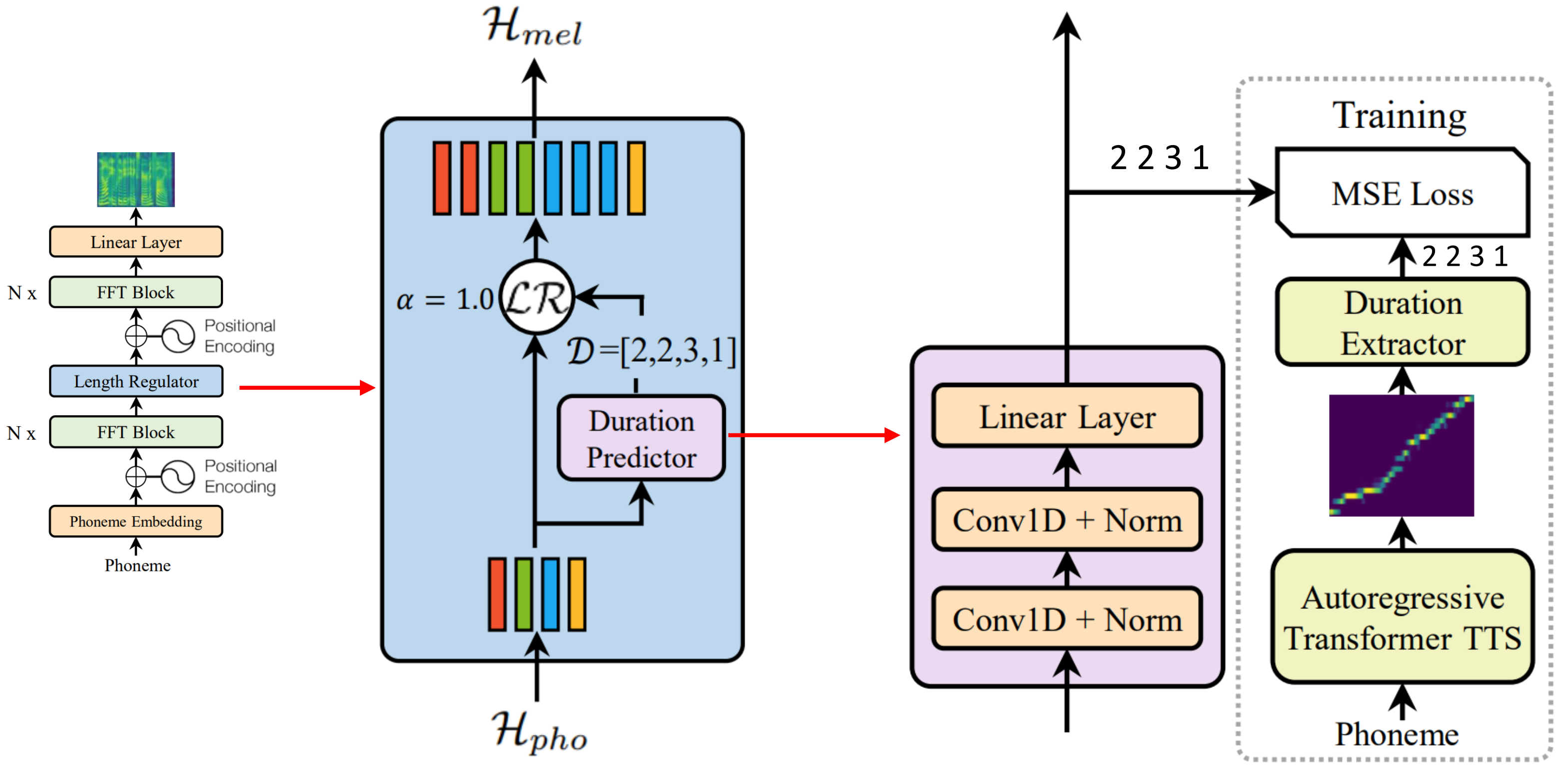
s p iy iy ch

0.5x speed — s p iy ch ← 2 2 3 1 (α = 2 (4 4 6 2))

s s s s p p p p iy iy iy iy iy iy ch ch

Inference speedup——FastSpeech

- Framework: Duration Predictor



- How to get the label to train the duration predictor?
- Extract duration based on the attention alignments from the autoregressive teacher

Inference speedup——FastSpeech

- FastSpeech has the following advantages
  - **Extremely fast**: **270x** inference speedup on mel-spectrogram generation, **38x** speedup on final waveform generation!
  - **Robust**: no bad case of words skipping and repeating.
  - **Controllable**: can control voice speed and prosody.
  - **Voice quality**: on par or better than previous SOTA model.

Inference speedup——FastSpeech

- Product Transfer: FastSpeech is deployed on Microsoft **Azure Speech Service (TTS)** for **54 languages/locales**

Languages	Locales	Languages	Locales	Languages	Locales	Languages	Locales
Arabic	ar-EG, ar-SA	Finnish	fi-FI	Japanese	ja-JP	Slovenian	sl-SI
Bulgarian	bg-BG	French	fr-FR, fr-CA, fr-CH	Korean	ko-KR	Spanish	es-ES, es-MX
Catalan	ca-ES	German	de-DE, de-AT, de-CH	Malay	ms-MY	Swedish	sv-SE
Chinese	zh-CN, zh-HK, zh-TW	Greek	el-GR	Norwegian	nb-NO	Tamil	ta-IN
Croatian	hr-HR	Hebrew	he-IL	Polish	pl-PL	Telugu	te-IN
Czech	cs-CZ	Hindi	hi-IN	Portuguese	pt-BR, pt-PT	Thai	th-TH
Danish	da-DK	Hungarian	hu-HU	Romanian	ro-RO	Turkish	tr-TR
Dutch	nl-NL	Indonesian	id-ID	Russia	ru-RU	Vietnamese	vi-VN
English	en-US, en-UK, en-AU, en-CA, en-IN, en-IE	Italian	it-IT	Slovak	sk-SK	Irish	ga-IE
Estonian	et-EE	Maltese	mt-MT	Lithuanian	lt-LT	Latvian	lv-LV

https://azure.microsoft.com/en-us/services/cognitive-services/text-to-speech

Inference speedup——FastSpeech 2

- The improvement space for FastSpeech
  - **Training pipeline complicated**: two-stage teacher-student distillation
  - **Target is not good**: the target mels distilled from teacher suffer from information loss
  - **Duration is not accurate**: the duration extracted from teacher is not accurate enough
- Improvements in FastSpeech 2 [27]
  - **Simplify training pipeline**: remove teacher-student distillation
  - **Use ground-truth speech as target**: avoid information loss
  - **Improve duration** & **Introduce more variance information**: ease the **one-to-many mapping** problem

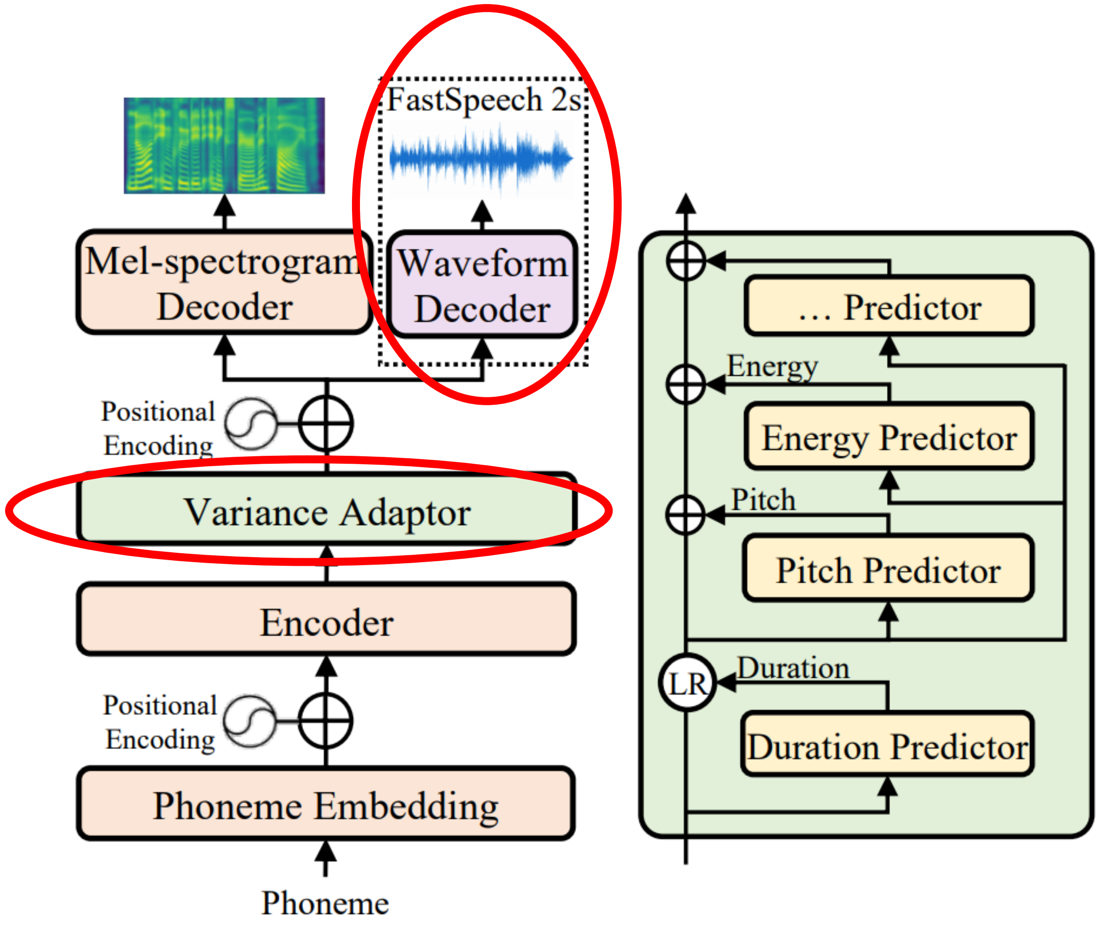
Text

↓

multiple speech variations

(duration, pitch, sound volume, speaker, style, emotion, etc)

Inference speedup——FastSpeech 2



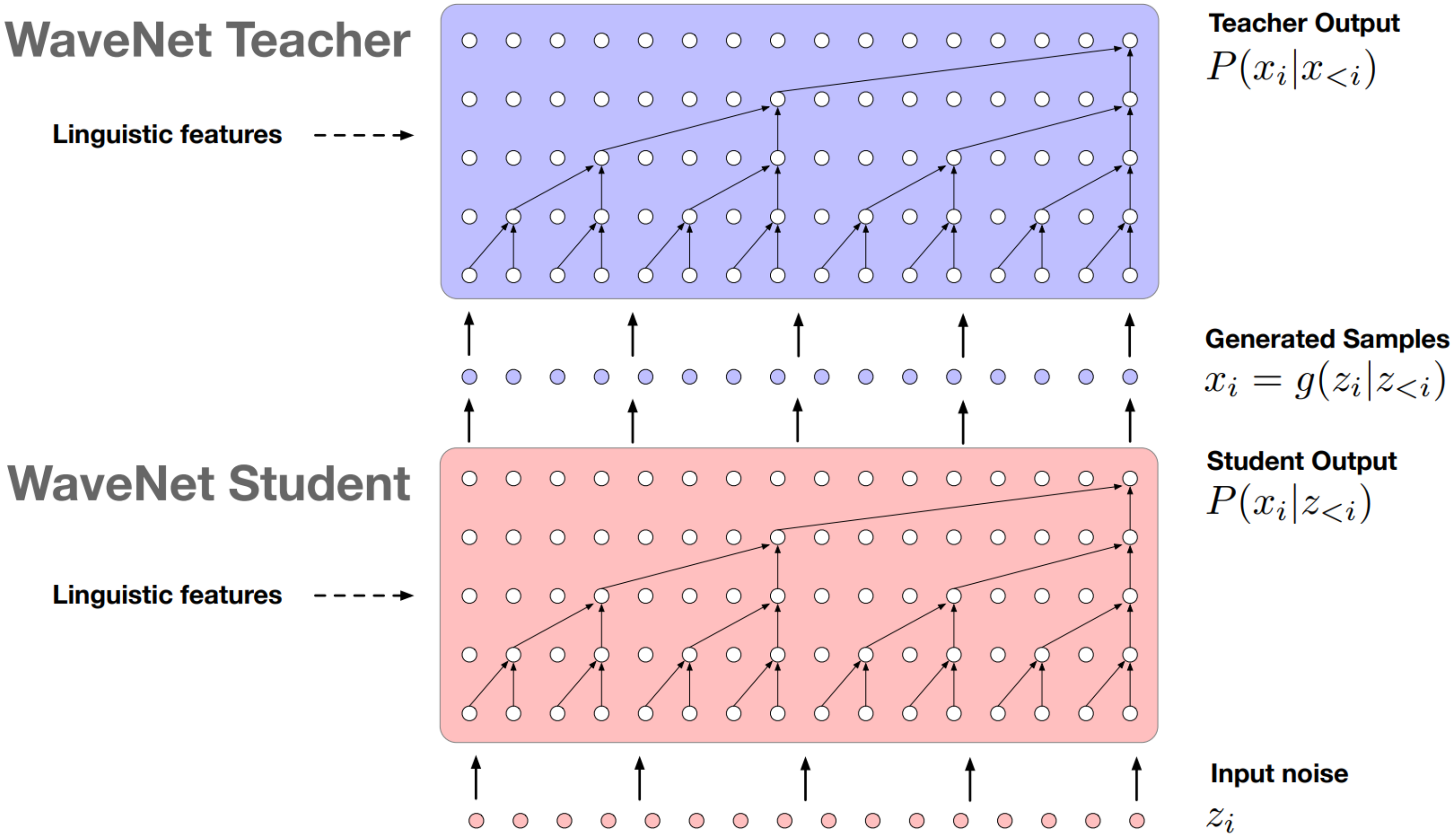
(a) FastSpeech 2

(b) Variance adaptor

- Variance adaptor: use variance predictor to predict duration, pitch, energy, etc.
- FastSpeech 2 improves FastSpeech with
  - more simplified training pipeline
  - higher voice quality
  - maintain the advantages of **fast, robust and even more controllable** synthesis in FastSpeech
- FastSpeech 2s
  - a fully end-to-end text to wave neural model
  - comparable (high) quality with FastSpeech 2

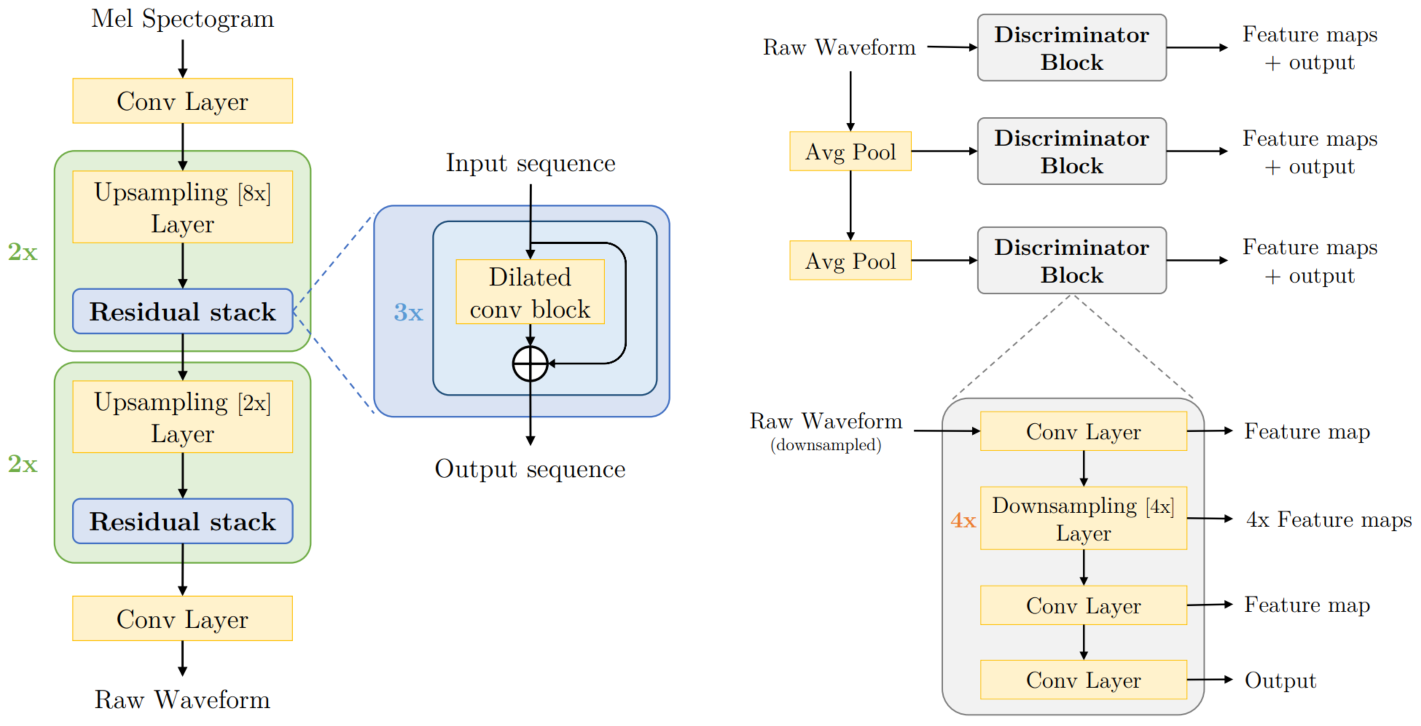
Inference speedup——Vocoder

- Parallel WaveNet [7]



Inference speedup——Vocoder

- GAN based model: MelGAN [15]
  - Generator: Transposed conv for upsampling, dilated conv to increase receptive field
  - Discriminator: Multi-scale discrimination



(a) Generator (b) Discriminator

Inference speedup——Vocoder

- Flow based model: WaveGlow [11]
  - Flow based transformation

$$z \sim \mathcal{N}(z; 0, I)$$

$$x = f_0 \circ f_1 \circ \ldots f_k(z)$$

$$\log p_\theta(x) = \log p_\theta(z) + \sum_{i=1}^{k} \log|\det(J(f_i^{-1}(x)))|$$

$$z = f_k^{-1} \circ f_{k-1}^{-1} \circ \ldots f_0^{-1}(x)$$

  - Affine Coupling Layer

$$x_a, x_b = split(x)$$

$$(\log s, t) = WN(x_a, mel\text{-}spectrogram)$$

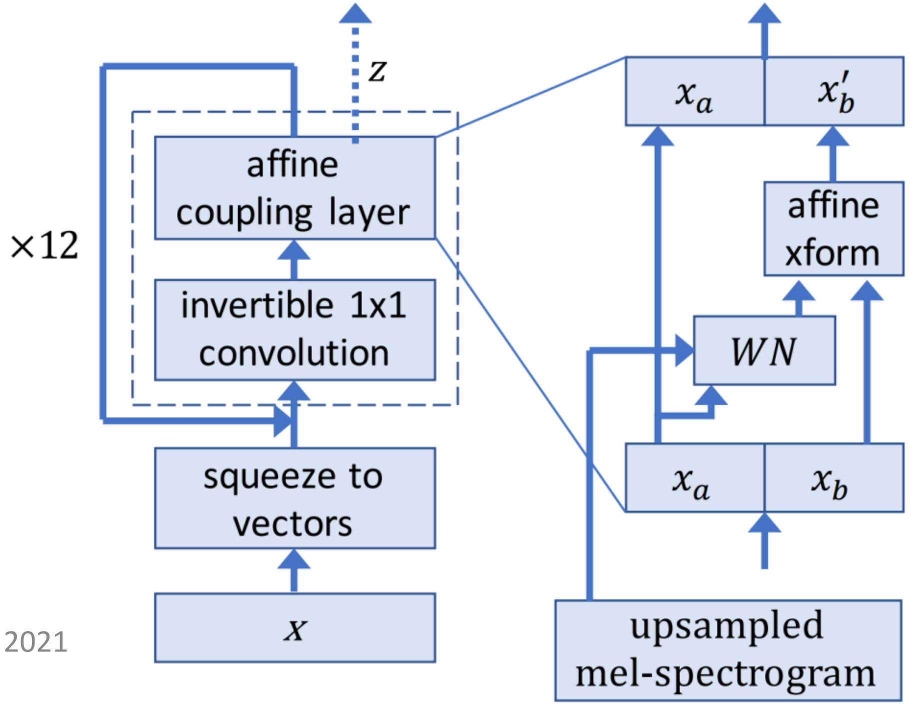
$$x_b\prime = s \odot x_b + t$$

$$f_{coupling}^{-1}(x) = concat(x_a, x_b\prime)$$

  - 1x1 Invertible Convolution

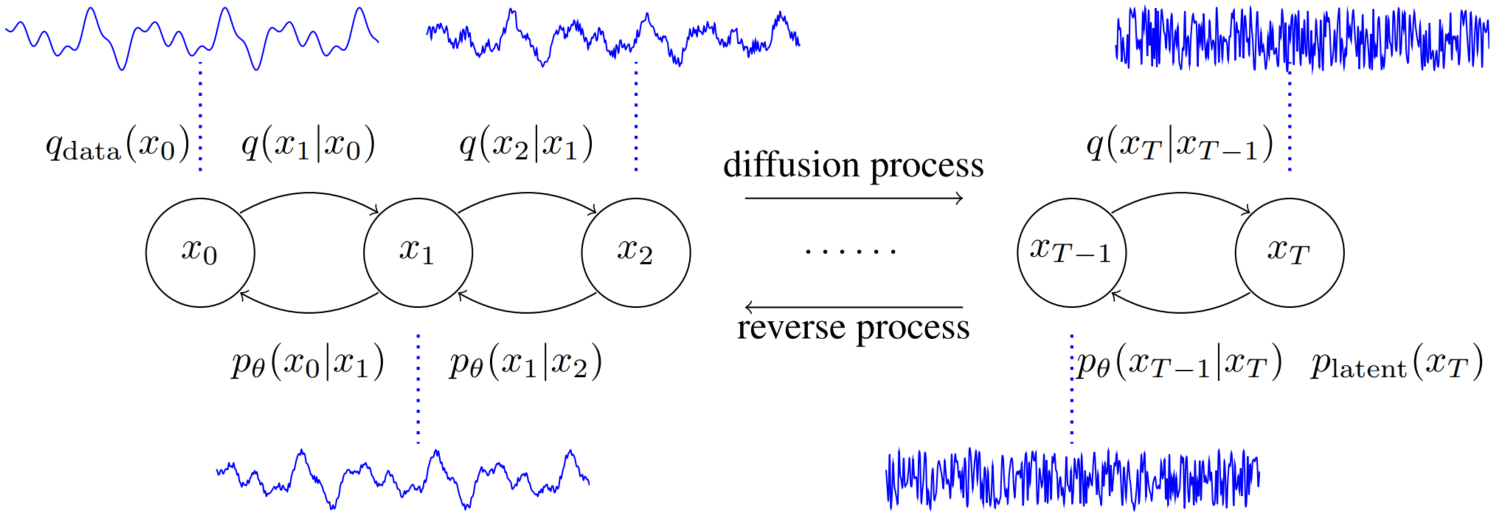
$$f_{conv}^{-1} = Wx$$

$$\log|\det(J(f_{conv}^{-1}(x)))| = \log|\det W|$$



Inference speedup——Vocoder

- Diffusion probabilistic model: DiffWave [31], WaveGrad [32]



Algorithm 1 Training

for $i = 1, 2, \cdots, N_{\text{iter}}$ **do**
  Sample $x_0 \sim q_{\text{data}}$, $\epsilon \sim \mathcal{N}(0, I)$, and
    $t \sim \text{Uniform}(\{1, \cdots, T\})$
  Take gradient step on
    $\nabla_\theta \|\epsilon - \epsilon_\theta(\sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t}\epsilon,\ t)\|_2^2$
    according to Eq. (7)
end for

Algorithm 2 Sampling

Sample $x_T \sim p_{\text{latent}} = \mathcal{N}(0, I)$
for $t = T, T-1, \cdots, 1$ **do**
  Compute $\mu_\theta(x_t, t)$ and $\sigma_\theta(x_t, t)$ using Eq. (5)
  Sample $x_{t-1} \sim p_\theta(x_{t-1}|x_t) =$
    $\mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_\theta(x_t, t)^2 I)$
end for
return x_0

Inference speedup——Lightweight model

- WaveRNN [9]
  - RNN with dual softmax layer, weight pruning, subscale prediction
- LPCNet [10]
  - Combine DSP with NN, linear prediction coefficient, more lightweight model
- Multiband modeling: Multi-band WaveRNN/MelGAN [37,38]
  - Subband technique
- Model compression
  - Pruning, quantization, knowledge distillation, neural architecture search

Outline

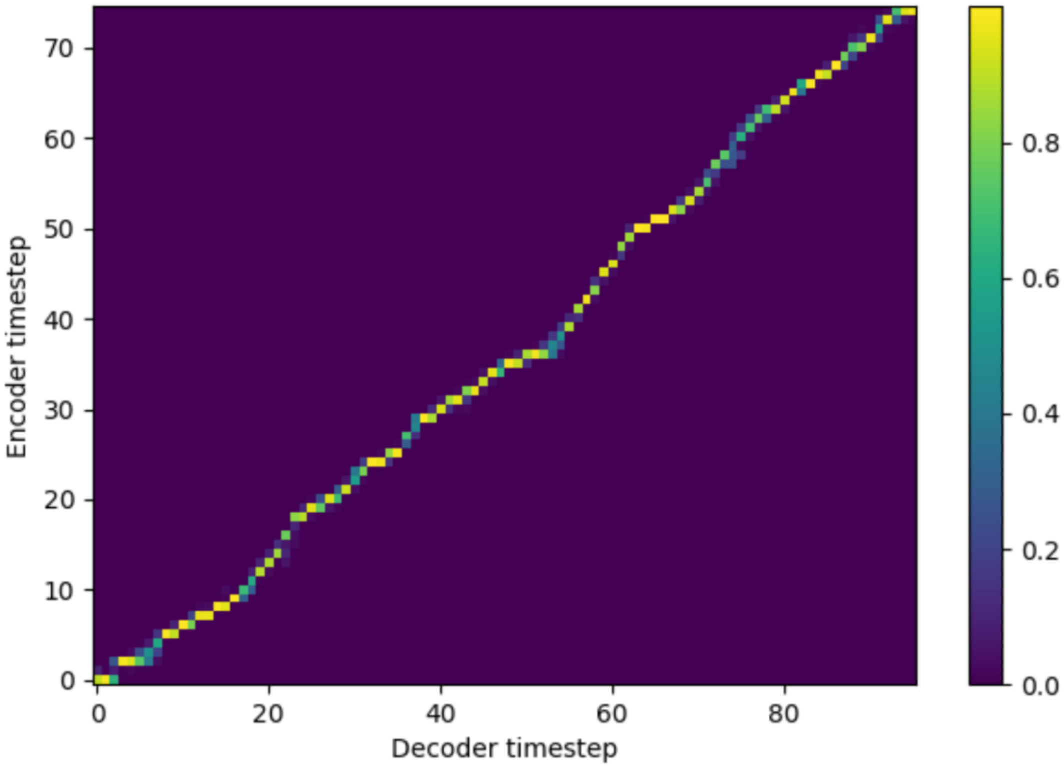
- Overview of text to speech
- Pushing the frontier of neural text to speech
  - More end-to-end
  - Inference speedup
  - Robustness, expressiveness and controllability
  - Low-resource
  - From research to product
- Summary

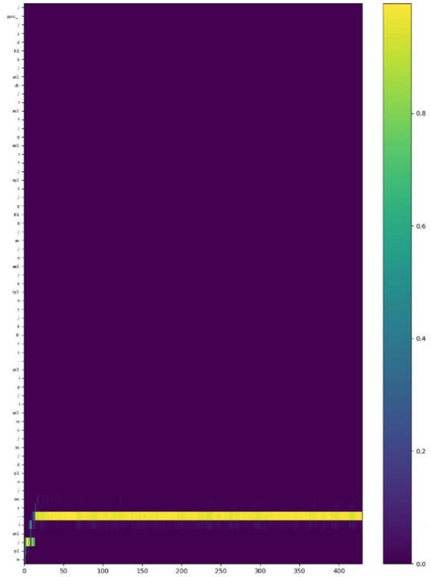
Robustness, expressiveness and controllability

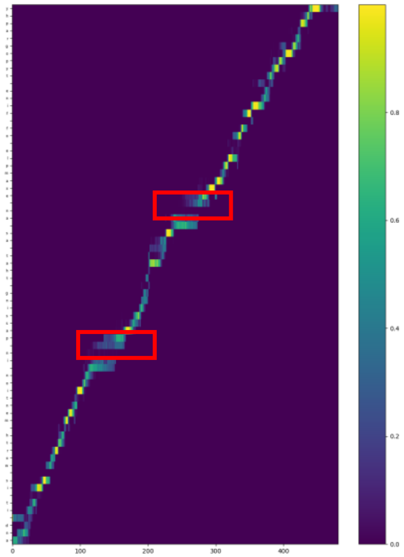
- Robustness
  - Attention improvement
  - Duration expansion
- Expressiveness
  - Over-smoothing prediction
  - Prosody modeling
- Controllability
  - Duration, pitch, energy, prosody, emotion, speaker, noise
  - Tag/label

Robustness——Attention improvement

- Encoder-decoder attention: Attention between mel-spectrogram and phoneme
  - Monotonic and diagonal



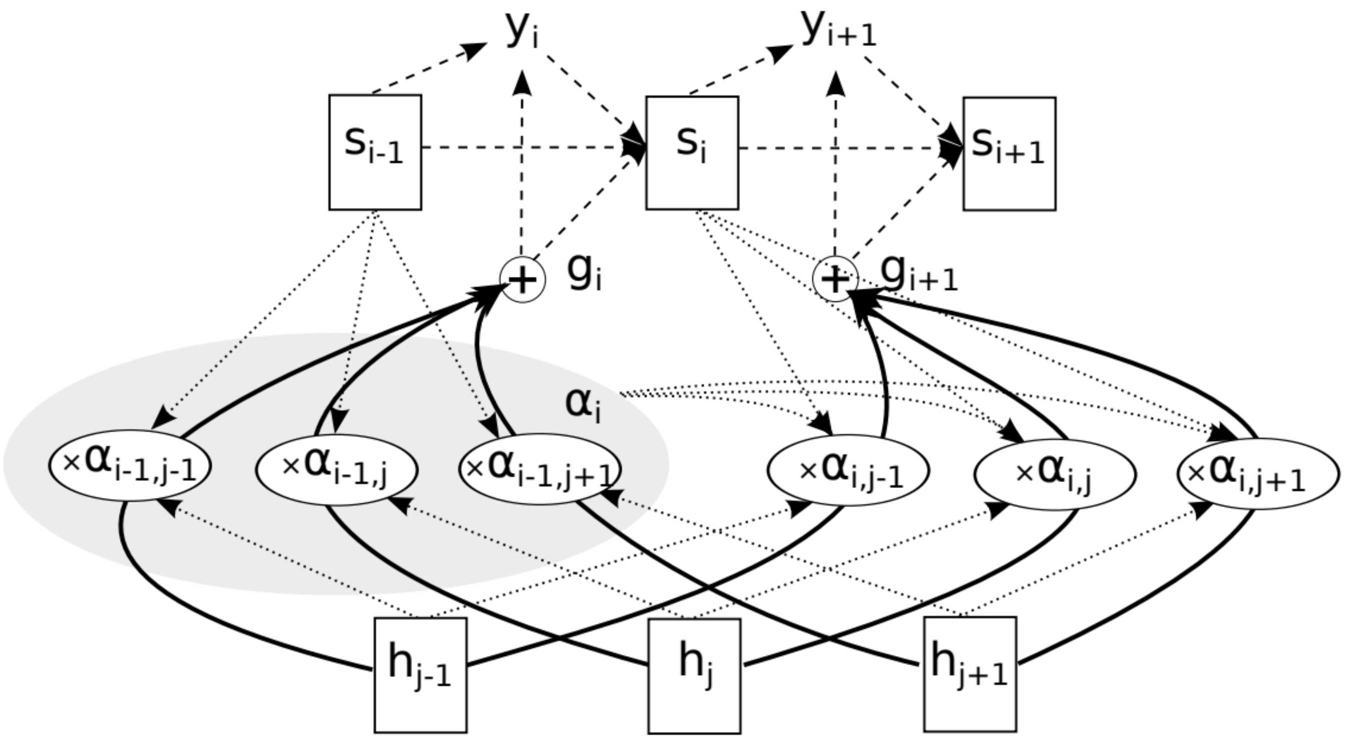




And it is worth mention in passing that, as an example of fine typography

Robustness——Attention improvement

- Location sensitive attention [39]
  - Use previous alignment to compute the next attention alignment



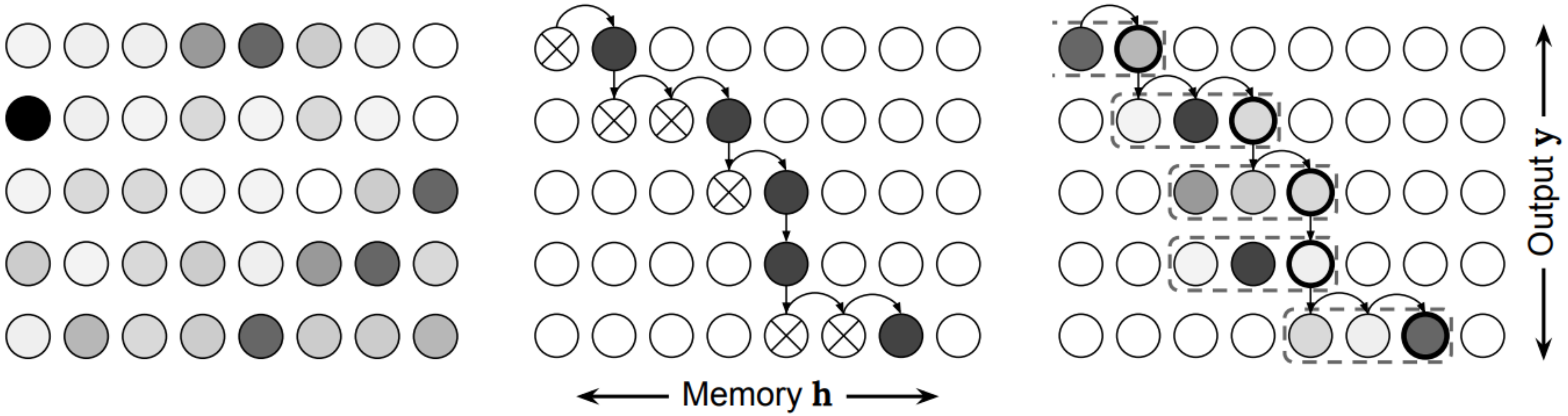
$$\alpha_i = Attend(s_{i-1}, \alpha_{i-1}, h)$$

$$g_i = \sum_{j=1}^{L} \alpha_{i,j} h_j$$

$$y_i \sim Generate(s_{i-1}, g_i),$$

Robustness——Attention improvement

- Monotonic attention [40]
  - The attention position is monotonically increasing



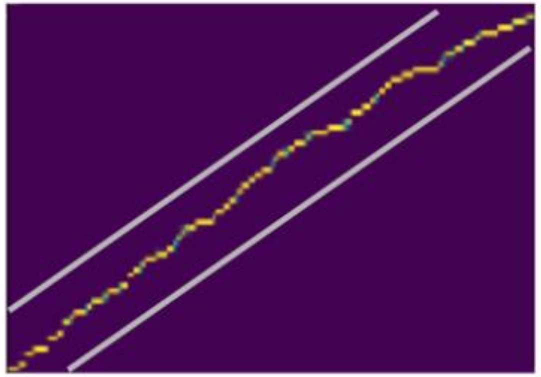
(a) Soft attention. (b) Hard monotonic attention. (c) Monotonic chunkwise attention.

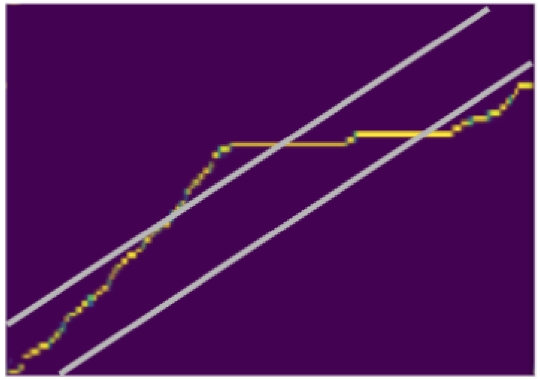
$$e_{i,j} = \mathrm{MonotonicEnergy}(s_{i-1}, h_j)$$
$$p_{i,j} = \sigma(e_{i,j})$$
$$z_{i,j} \sim \mathrm{Bernoulli}(p_{i,j})$$

Robustness——Attention improvement

- Windowing [41,42]
  - Only a subset of the encoding results $\hat{\boldsymbol{x}} = [\boldsymbol{x}_{p-w}, ..., \boldsymbol{x}_{p+w}]$ are considered at each decoder timestep when using the windowing technique [1] [2]

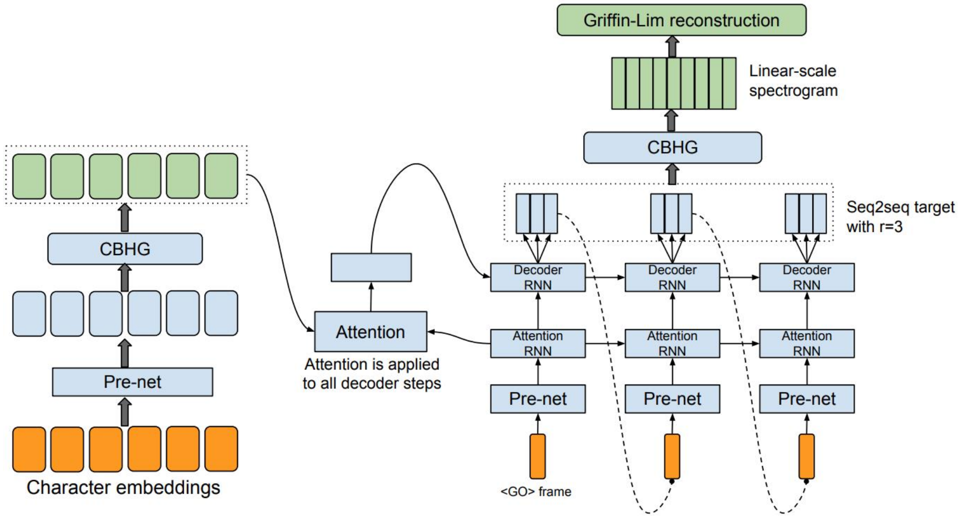
- Penalty loss for off-diagonal attention distribution [43]
  - Guided attention loss with diagonal band mask

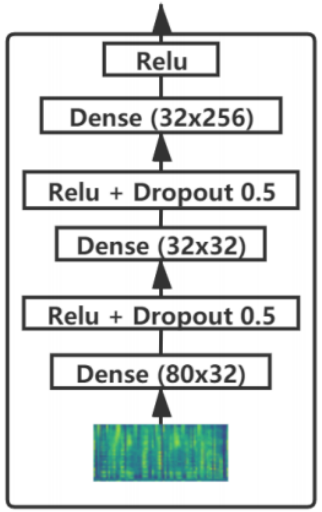




Robustness——Attention improvement

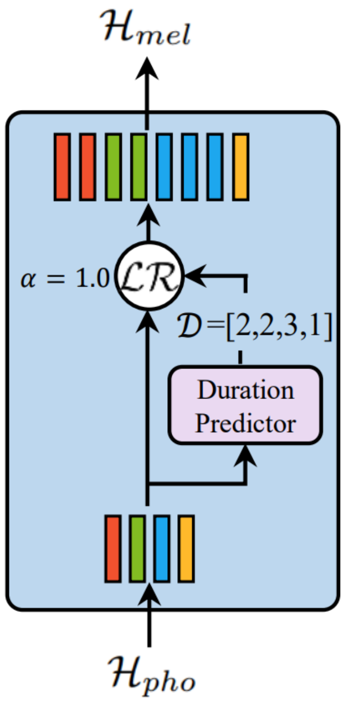
- Multi-frame prediction [21]
  - Predicting multiple, non-overlapping output frames at each decoder step
  - Increase convergence speed, with a much faster (and more stable) alignment learned from attention
- Decoder prenet dropout/bottleneck [21,43]
  - 0.5 dropout, small hidden size as bottleneck



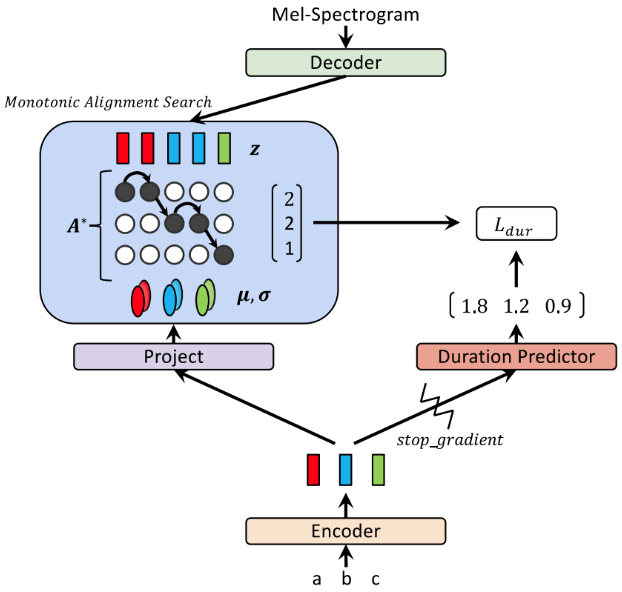


Robustness——Duration Prediction

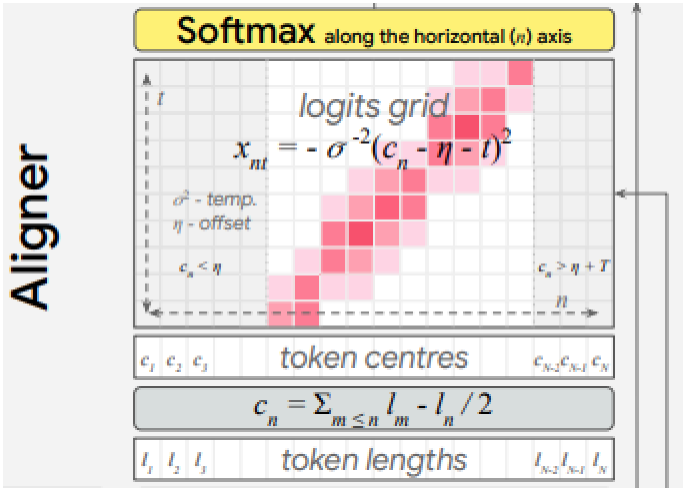
- Duration prediction and expansion
  - SPSS → Seq2Seq model with attention → Non-autoregressive model
  - Duration → attention, no duration → duration prediction (technique renaissance!)



FastSpeech [26]



Glow-TTS [30]



EATS [28]

Expressiveness——Over-smoothness

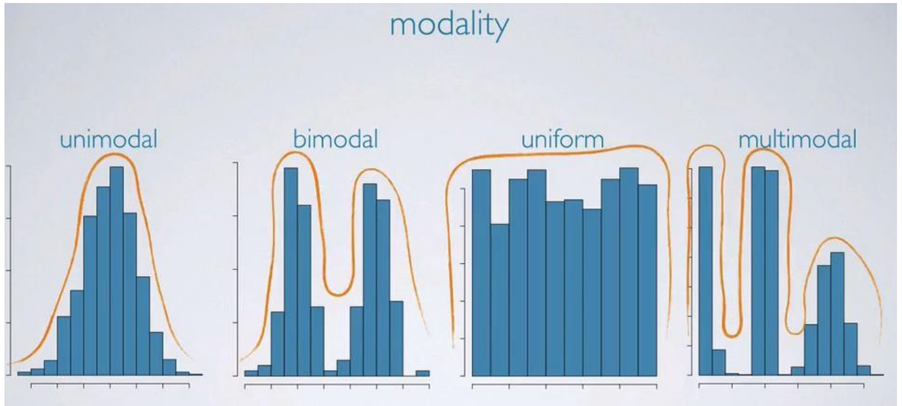
- Over-smoothing prediction
  - One to many mapping in text to speech: p(y|x) multimodal distribution

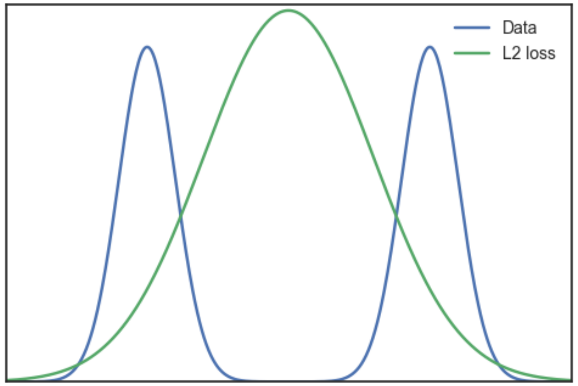
Text

↓

multiple speech variations

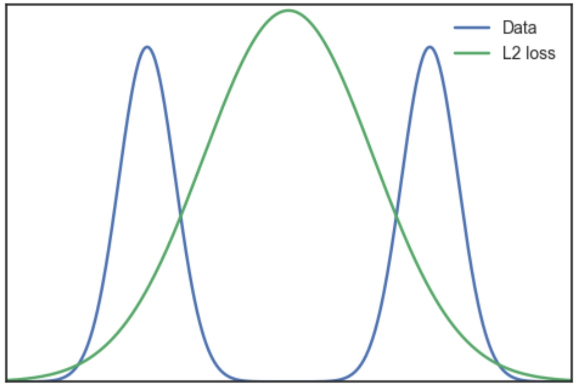
(duration, pitch, sound volume, speaker, style, emotion, etc)





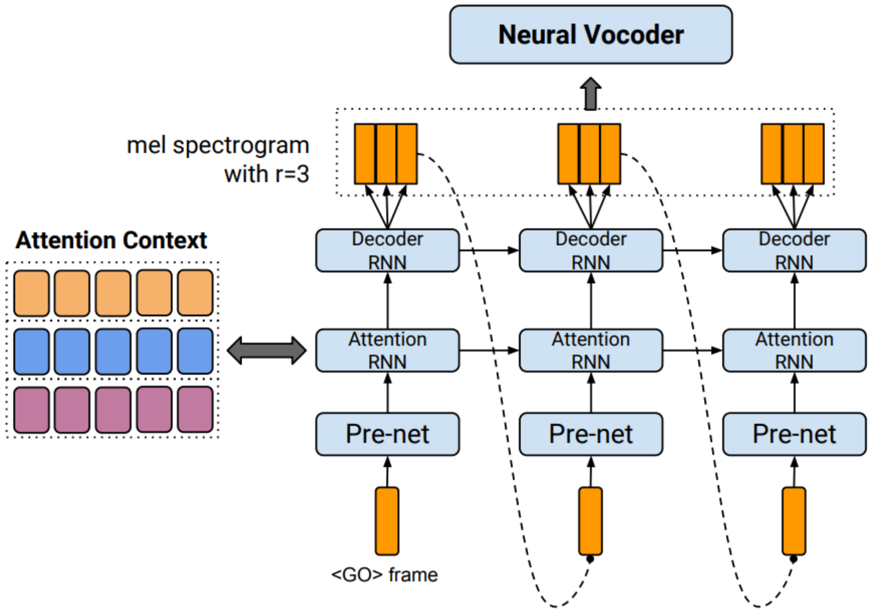
Expressiveness——Over-smoothness

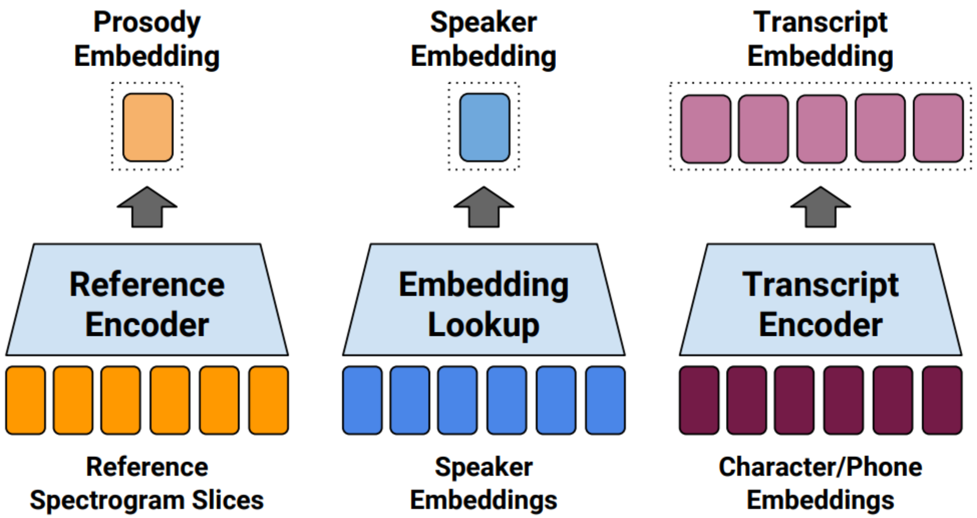
- How to solve over-smoothness
  - Simplify input-output distribution p(y|x)
    - More input information: Pitch, duration, energy, speaker ID, prosody tag, etc..
    - Simplify target: Data distillation: lossy, Data transformation: Short Time Fourier Transformation (STFT), DCT, Wavelet
  - More advanced loss for multimodal modeling
    - L1: Laplace distribution [44,45], L2: Gaussian distribution
    - Mixture of Gaussian/Laplace/Logistic: multimodal distribution
    - High-order statistics loss: high-order moment, SSIM
    - Model-based loss (any distribution): classifier, discriminator in GAN



Expressiveness——Prosody modeling

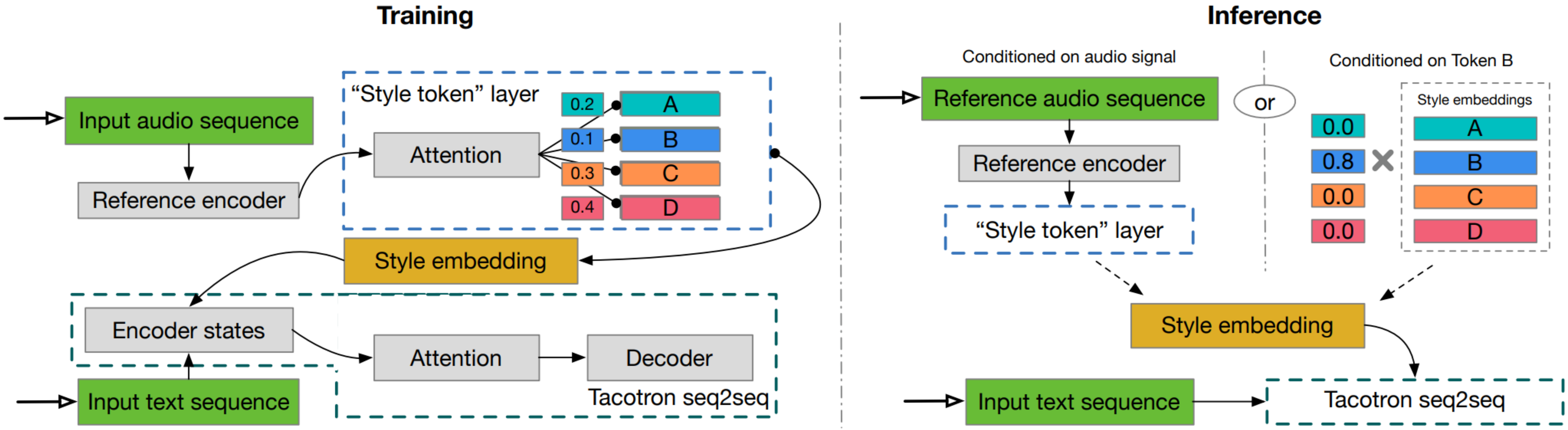
- Prosody embedding from reference audio [47]





Expressiveness——Prosody modeling

- Prosody embedding from reference audio [47]
- Prosody embedding from style tokens [46]

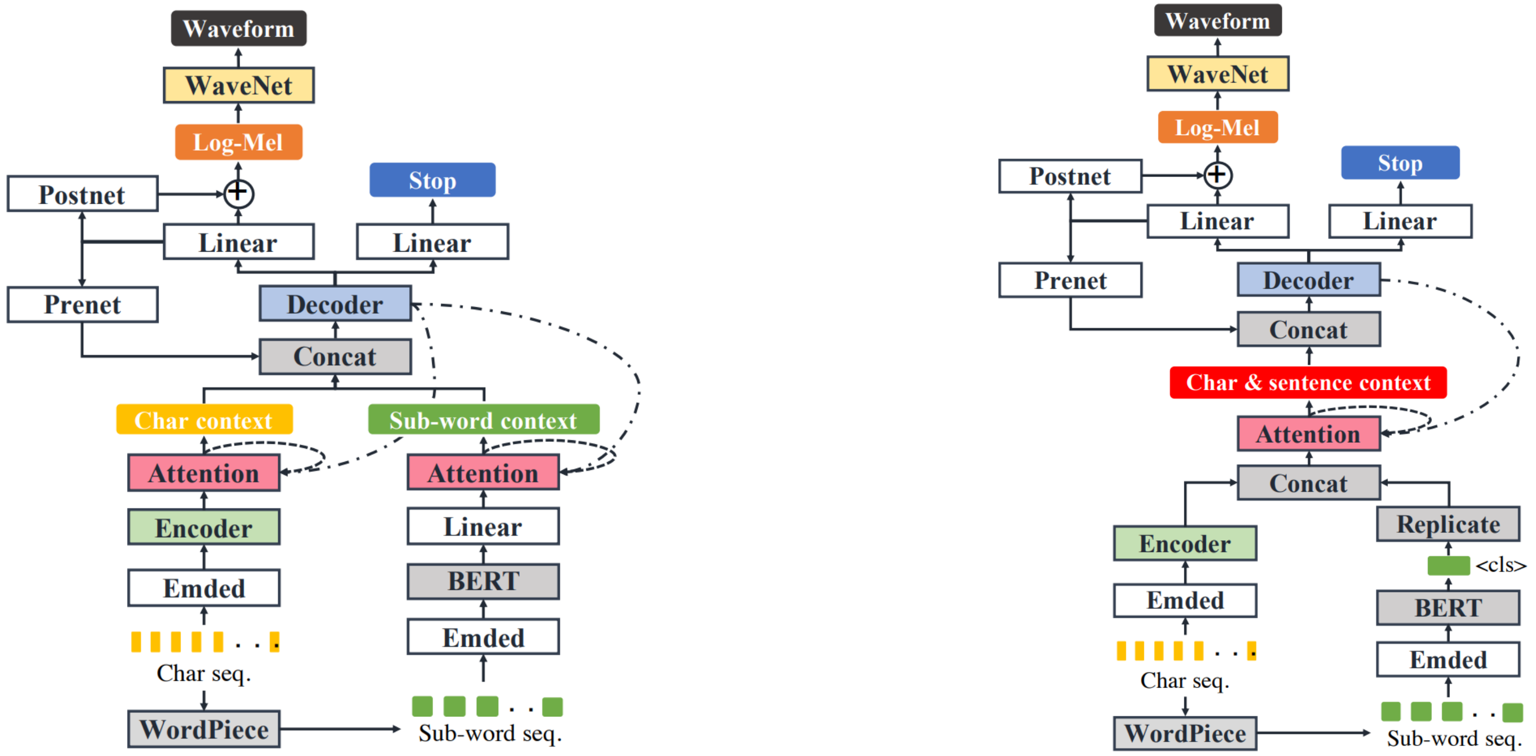


Expressiveness——Prosody modeling

- Prosody embedding from reference audio [47]
- Prosody embedding from style tokens [46]
- Prosody embedding from different granularities
  - Frame-level, phoneme-level, syllable-level, word-level, utterance-level, speaker-level [48,49,50,51,52]

Expressiveness——Pre-training

- Text pre-training, e.g., BERT [53,54,55]

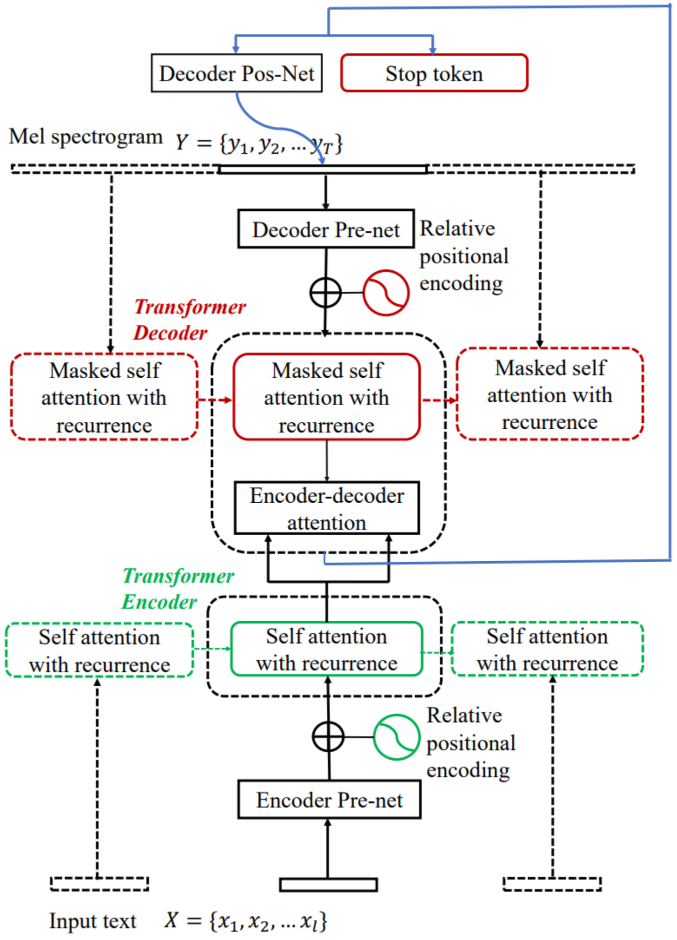


(a) *Subword-level model*

(b) *Phrase-level model*

Expressiveness——Long-form/paragraph

- Leverage contextual (before and after) sentences for prosody modeling [71]

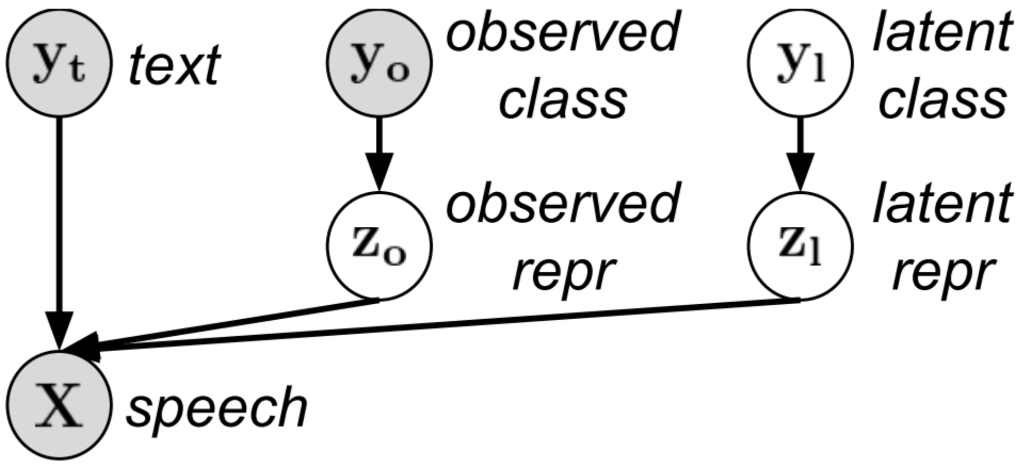


Controllability

- What attributes to control
  - Duration, pitch, energy, prosody, emotion, speaker, noise, etc
- Control with attribute value/tag
  - Train with tag as input, inference use corresponding tag to control
  - Duration value, or speed tag (slow/fast), F0/energy value, speaker embedding, reference audio, style tokens, emotion tag, noise tag, etc
- However, when no tag/label available, or only part available
  - How to disentangle and control the attributes is challenging

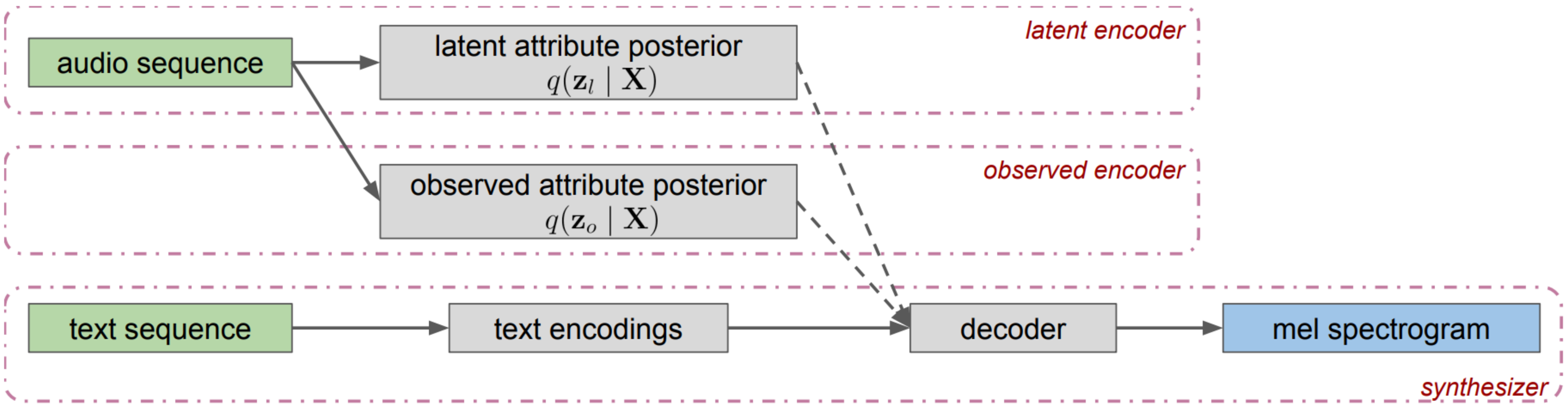
Controllability——Semi-supervised

- VAE model [56]
  - Observed: labeled attributes
  - Latent: unlabeled attributes
- Partial supervision to the latent variables of VAE
  - With only 1% label data, to control affect or speaking rate



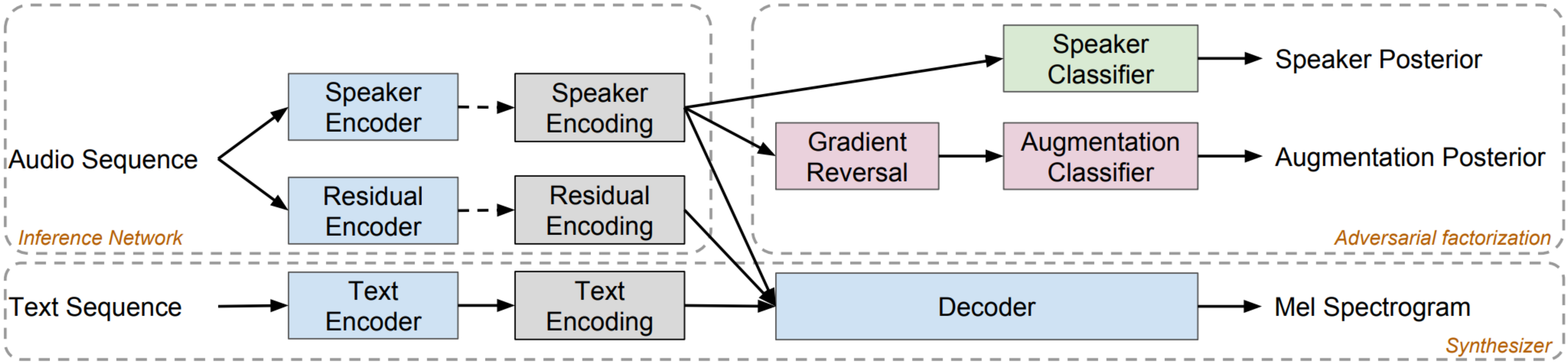
Controllability——Disentanglement

- GMVAE-Tacotron [57]
  - Mixture parameters can be analyzed to understand what each component corresponds to, similar to GST



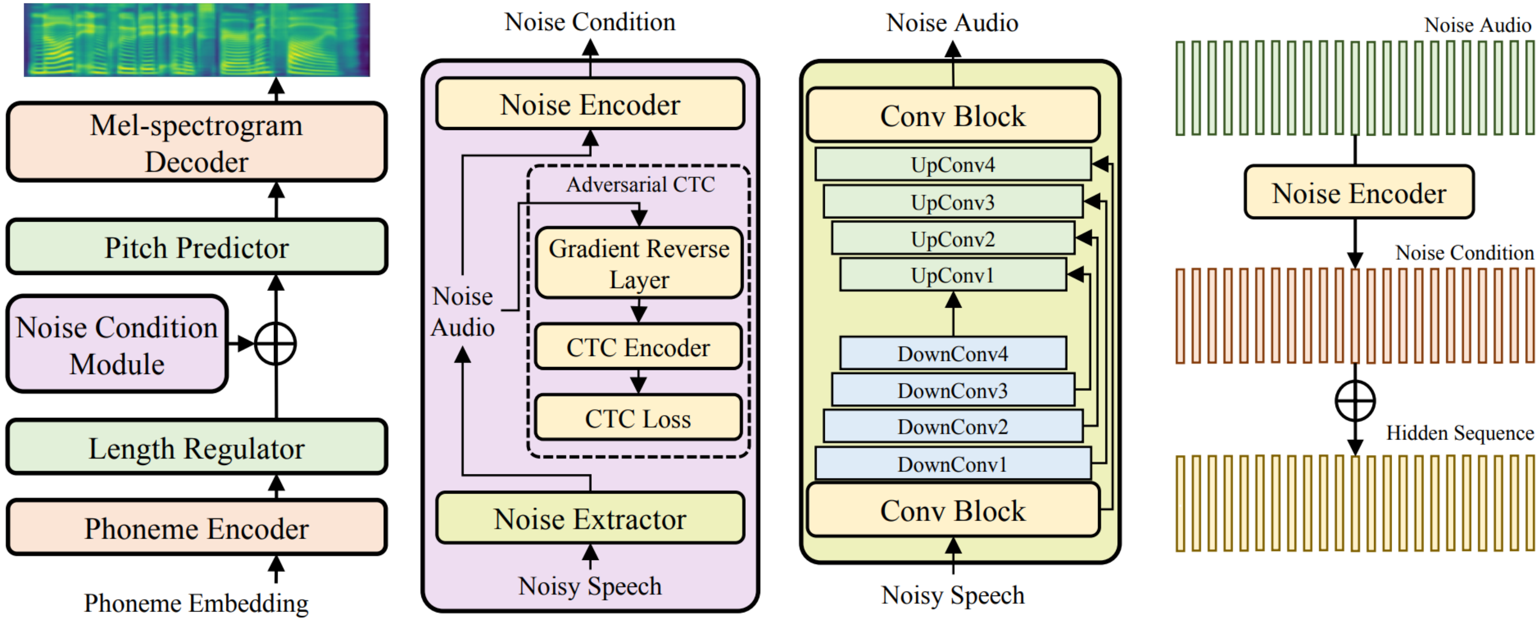
Controllability——Denoising

- Disentangling correlated speaker and noise [58]
  - Synthesize clean speech for noisy speakers



Controllability——Denoising

- Disentangling correlated speaker and noise with frame-level modeling [59]
  - Synthesize clean speech for noisy speakers



(a) DenoiSpeech (b) Noise Condition Module (c) Noise Extractor (d) Noise Encoder

Outline

- Overview of text to speech
- Pushing the frontier of neural text to speech
  - More end-to-end
  - Inference speedup
  - Robustness, expressiveness and controllability
  - Low-resource
  - From research to product
- Summary

Low-resource TTS

- There are **7,000+** languages in the world, but popular commercialized speech services only support **dozens of** languages

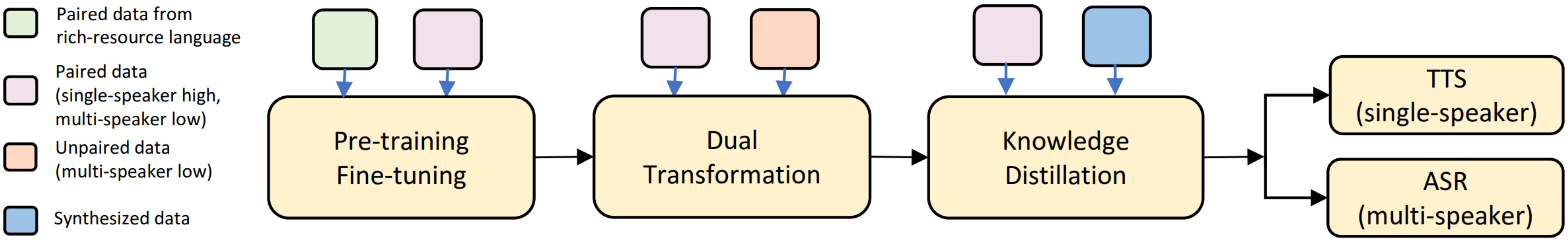
	Azure Speech Service: TTS	Azure Speech Service: ASR	Windows	World
#languages	50+	40+	200+	7000+

- There is strong business demand to support more languages in TTS. However, the data collection cost is high.
  - For TTS, the minimum data labeling cost for one language: ¥ 1 million

Low-resource TTS

- Techniques for low-resource TTS
    - Cross-lingual pre-training, paired data [61,72]
    - Mono-lingual pre-training, unpaired text or speech [62,63,69]
    - TTS ←→ASR, Speech Chain, Dual Learning, Cycle Consistency [60,61,64,65]

Low-resource TTS——LRSpeech [61]



- **Step 1**: Language transfer
  - Human languages share similar pronunciations; Rich-resource language data is "free"
- **Step 2**: TTS and ASR help with each other
  - Leverage the task duality with unpaired speech and text data
- **Step 3**: Customization for product deployment with knowledge distillation
  - Better accuracy by data knowledge distillation
  - Customize multi-speaker TTS to a target-speaker TTS, and to small model

Low-resource TTS——LRSpeech

- Results

Language	Intelligibility Rate (IR)	Mean Opinion Score (MOS)
English	98.08	3.57
Lithuanian	98.60	3.65

LRSpeech achieves **high IR score (>98%)** and **MOS score (>3.5)**

- Data cost

Data Resource	Full-Resource	Speech Chain [36]	Almost Unsup [29]	SeqRQ-AE [20]	Our Method
Text normalization rule	✓	?	✓	✓	✓
Pronunciation lexicon	✓	×	✓	✓	×
Paired data (single-speaker, high)	dozens of hours	20 hours	200 sentences	200 sentences	50 sentences
Paired data (multi-speaker, low)	hundreds of hours	×	×	×	1000 sentences
Unpaired speech (single-speaker, high)	×	80 hours	13000 sentences	13000 sentences	×
Unpaired speech (multi-speaker, low)	×	×	×	×	13000 sentences
Unpaired text	×	✓	✓	✓	✓
Total Data Cost	312000	120000	74000	74000	833

LRSpeech

100x data cost reduction compared to previous works

Low-resource TTS——LRSpeech

- Product deployment
  - LRSpeech has been deployed in Microsoft Azure Text to Speech service
  - Extend 5 new low-resource languages for TTS: Irish, Lithuanian, Latvian, Estonian, Maltese

Locale	Language (Region)	Average MOS	Intelligibility
mt-MT	Maltese (Malta)	3.59*	98.40%
lt-LT	Lithuanian (Lithuania)	4.35	99.25%
et-EE	Estonian (Estonia)	4.52	98.73%
ga-IE	Irish (Ireland)	4.62	99.43%
lv-LV	Latvian (Latvia)	4.51	99.13%

https://techcommunity.microsoft.com/t5/azure-ai/neural-text-to-speech-previews-five-new-languages-with/ba-p/1907604

Outline

- Overview of text to speech
- Pushing the frontier of neural text to speech
  - More end-to-end
  - Inference speedup
  - Robustness, expressiveness and controllability
  - Low-resource
  - From research to product
- Summary

From research to product

- Difference between research and product deployment

Research	Product
Non-trivial and useful: Novelty, deep investigation on non-trivial solutions	Practically useful: Even if not novel or non-trivial
Advantages in principle and in experiment results	99.99% usability, but not cherry-pick good cases
Story driven	Practical deployment

- More difficult to solve a product problem than publish a paper
  - Maybe just need 3 months to rush a good paper, but takes 1 year to ship it into product
  - However, research has great value and is irreplaceable
  - We just need to take practical usage into consideration during research

From research to product——Custom voice

- Background
  - Custom Voice is an important service in text to speech
  - Microsoft Azure: https://speech.microsoft.com/customvoice
  - Amazon AWS: https://aws.amazon.com/polly/
  - Google Cloud: https://cloud.google.com/text-to-speech/custom-voice/docs
- The scenario is to support TTS for the voice of any user/customer
  - User need record their voice with few sentences using their own devices
  - Upload to speech service for voice adaption
  - Speech service provide a custom model and serve for this voice

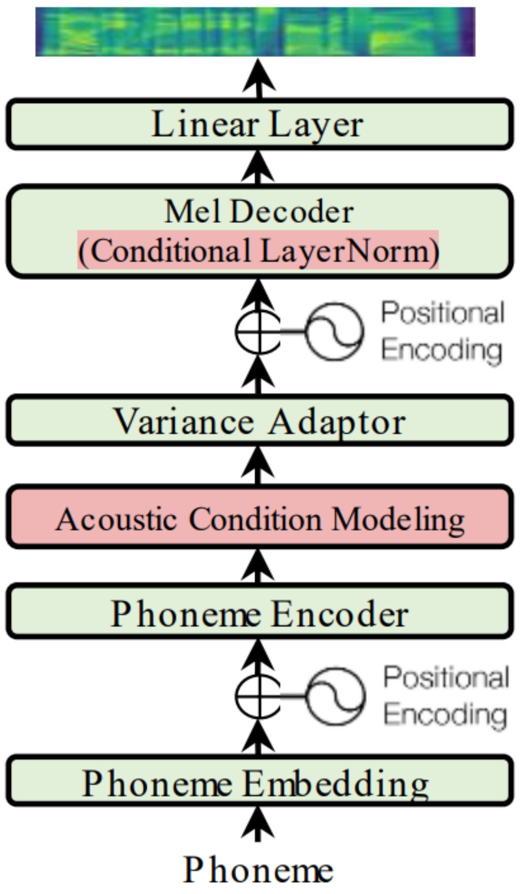
From research to product——Custom voice

- Challenges
  - To support diverse customers, the adaptation model needs to handle diverse acoustic conditions which are very different from source speech data
  - To support many customers, the adaptation parameters need to be small enough for each target speaker to reduce memory usage while maintaining high voice quality
    - e.g., each user/voice with 100MB, 1M users, total memory storage = 100PB!
- However, related works [66,67,68]
  - Too many adaptation parameters
  - Poor adaptation quality with few parameters
  - Only consider source and adaptation data are in the same domain

From research to product——Custom voice

- AdaSpeech [52]
  - Pre-training; Fine-tuning; Inference
  - Built on popular non-autoregressive TTS model, FastSpeech

  - Acoustic condition modeling
    - Model diverse acoustic conditions at speaker/utterance/phoneme level
  - Conditional layer normalization
    - To fine-tune as small parameters as possible while ensuring the adaptation quality
  - Consider adaptation data is different from source data
    - More challenging but close to product scenario



From research to product——Custom voice

Metric	Setting	# Params/Speaker	LJSpeech	VCTK	LibriTTS
MOS	*GT*	/	3.98 ± 0.12	3.87 ± 0.11	3.72 ± 0.12
	GT mel + Vocoder	/	3.75 ± 0.10	3.74 ± 0.11	3.65 ± 0.12
	Baseline (spk emb)	256 (256)	2.37 ± 0.14	2.36 ± 0.10	3.02 ± 0.13
	Baseline (decoder)	14.1M (14.1M)	3.44 ± 0.13	3.35 ± 0.12	3.51 ± 0.11
	AdaSpeech	1.2M (4.9K)	3.45 ± 0.11	3.39 ± 0.10	3.55 ± 0.12
SMOS	*GT*	/	4.36 ± 0.11	4.44 ± 0.10	4.31 ± 0.07
	GT mel + Vocoder	/	4.29 ± 0.11	4.36 ± 0.11	4.31 ± 0.07
	Baseline (spk emb)	256 (256)	2.79 ± 0.19	3.34 ± 0.19	4.00 ± 0.12
	Baseline (decoder)	14.1M (14.1M)	3.57 ± 0.12	3.90 ± 0.12	4.10 ± 0.10
	AdaSpeech	1.2M (4.9K)	3.59 ± 0.15	3.96 ± 0.15	4.13 ± 0.09

1. vs Baseline (spk emb), AdaSpeech achieves better MOS and SMOS with similar parameters
2. vs Baseline (decoder), AdaSpeech achieves on par MOS and SMOS with much smaller adaptation parameters

From research to product

- Improve intelligibility, naturalness, robustness, expressiveness, controllability
  - Maybe not fully end-to-end, but need to be accurate, text normalization, grapheme-to-phoneme conversion are necessary
  - Avoid bad cases such as glitches, hoarseness, metallic noise, jitter, pitch break, etc
  - Long-form/paragraph/narrative reading with emotion
- Reduce development cost
  - A universal multi-lingual/multi-speaker/multi-style TTS model, and fine-tune to any product scenarios
  - Small latency, memory, computation for deployment, especially in edge devices
  - Data efficiency, high quality with few data
- Extended product scenarios
  - Singing voice synthesis
  - Talking face synthesis

Outline

- Overview of text to speech
- Pushing the frontier of neural text to speech
  - More end-to-end
  - Inference speedup
  - Robustness, expressiveness and controllability
  - Low-resource
  - From research to product
- Summary

Summary

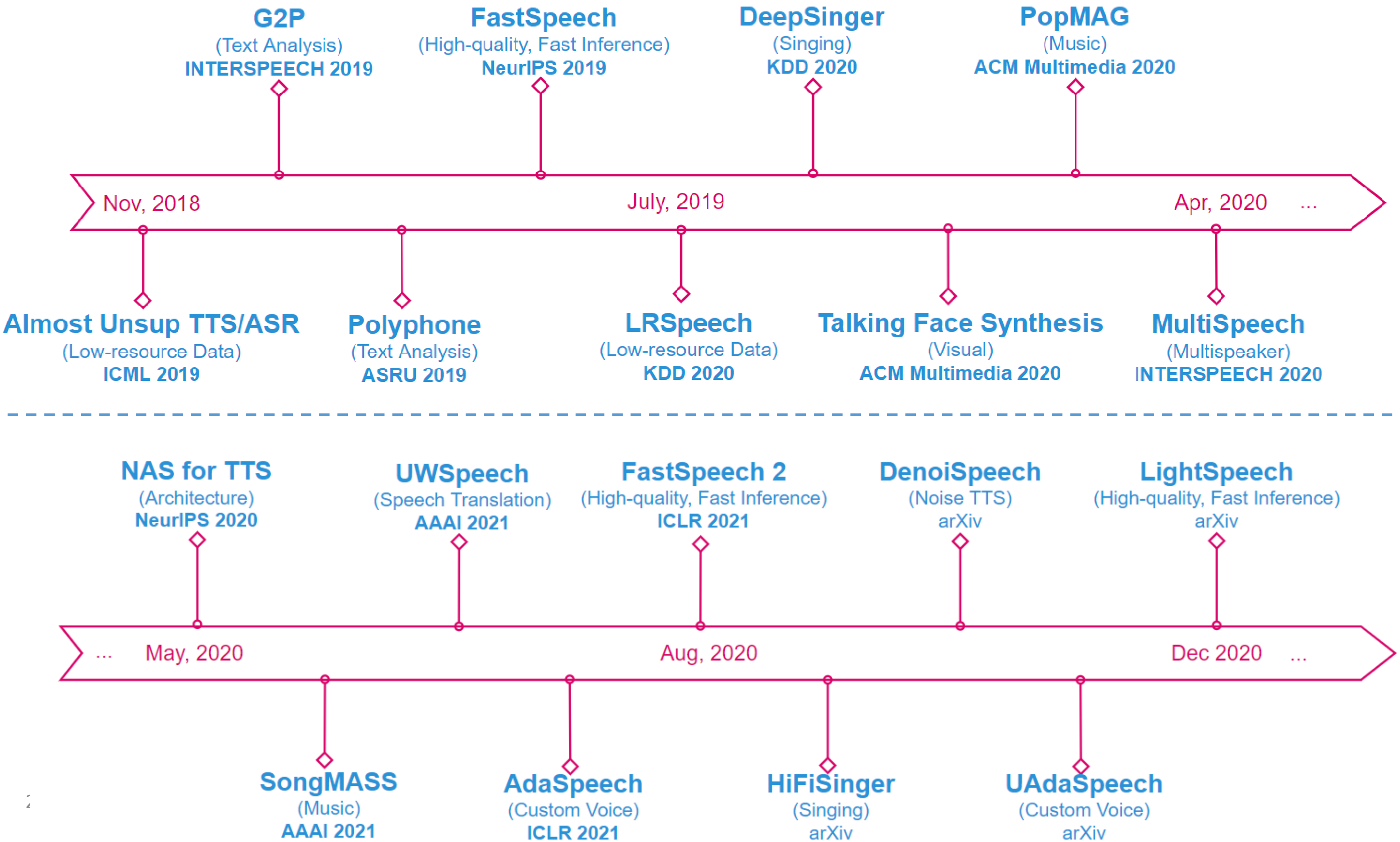
- TTS technology evolves from concatenative synthesis, statistical parametric synthesis, and neural based end-to-end synthesis
- Mainstream TTS model uses separate acoustic model and vocoder, but fully end-to-end TTS model is on the way
- Improving the quality while reducing the cost is always the goal of TTS
  - Quality: Intelligibility, naturalness, robustness, expressiveness and controllability
  - Cost: Engineering cost (end-to-end), serving cost (inference speedup), data cost (low resource)
- Research is the engine for TTS improvement, at the same time the engine should take practical usage into consideration

Thank You!

Xu Tan
Senior Researcher @ Microsoft Research Asia
xuta@microsoft.com

https://www.microsoft.com/en-us/research/people/xuta/
https://speechresearch.github.io/

Our research on speech



Reference

[1] Yoshimura, T., Tokuda, K., Masuko, T., Kobayashi, T., & Kitamura, T. (1999). Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis. In Sixth European Conference on Speech Communication and Technology.

[2] Ze, H., Senior, A., & Schuster, M. (2013, May). Statistical parametric speech synthesis using deep neural networks. In 2013 ieee international conference on acoustics, speech and signal processing (pp. 7962-7966). IEEE.

[3] Zen, H. (2015). Acoustic modeling in statistical parametric speech synthesis-from HMM to LSTM-RNN.

[4] Morise, M., Yokomori, F., & Ozawa, K. (2016). WORLD: a vocoder-based high-quality speech synthesis system for real-time applications. IEICE TRANSACTIONS on Information and Systems, 99(7), 1877-1884.

[5] Kawahara, H., Masuda-Katsuse, I., & De Cheveigne, A. (1999). Restructuring speech representations using a pitch-adaptive time–frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds. Speech communication, 27(3-4), 187-207.

[6] van den Oord, A., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., ... & Kavukcuoglu, K. WaveNet: A Generative Model for Raw Audio. In 9th ISCA Speech Synthesis Workshop (pp. 125-125).

[7] Oord, A., Li, Y., Babuschkin, I., Simonyan, K., Vinyals, O., Kavukcuoglu, K., ... & Hassabis, D. (2018, July). Parallel wavenet: Fast high-fidelity speech synthesis. In International conference on machine learning (pp. 3918-3926). PMLR.

[8] Mehri, S., Kumar, K., Gulrajani, I., Kumar, R., Jain, S., Sotelo, J., ... & Bengio, Y. (2016). SampleRNN: An unconditional end-to-end neural audio generation model. arXiv preprint arXiv:1612.07837.

[9] Kalchbrenner, N., Elsen, E., Simonyan, K., Noury, S., Casagrande, N., Lockhart, E., ... & Kavukcuoglu, K. (2018). Efficient neural audio synthesis. arXiv preprint arXiv:1802.08435.

[10] Valin, J. M., & Skoglund, J. (2019, May). LPCNet: Improving neural speech synthesis through linear prediction. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 5891-5895). IEEE.

Reference

[11] Prenger, R., Valle, R., & Catanzaro, B. (2019, May). Waveglow: A flow-based generative network for speech synthesis. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 3617-3621). IEEE.

[12] Kim, S., Lee, S. G., Song, J., Kim, J., & Yoon, S. (2018). FloWaveNet: A generative flow for raw audio. arXiv preprint arXiv:1811.02155.

[13] Ping W, Peng K, Zhao K, et al. Waveflow: A compact flow-based model for raw audio[C]//International Conference on Machine Learning. PMLR, 2020: 7706-7716.

[14] Donahue, C., McAuley, J., & Puckette, M. (2018). Adversarial audio synthesis. arXiv preprint arXiv:1802.04208.

[15] Kumar, K., Kumar, R., de Boissiere, T., Gestin, L., Teoh, W. Z., Sotelo, J., ... & Courville, A. C. (2019). Melgan: Generative adversarial networks for conditional waveform synthesis. In Advances in Neural Information Processing Systems (pp. 14910-14921).

[16] Yamamoto, R., Song, E., & Kim, J. M. (2020, May). Parallel WaveGAN: A fast waveform generation model based on generative adversarial networks with multi-resolution spectrogram. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 6199-6203). IEEE.

[17] Bińkowski, M., Donahue, J., Dieleman, S., Clark, A., Elsen, E., Casagrande, N., ... & Simonyan, K. (2019). High fidelity speech synthesis with adversarial networks. arXiv preprint arXiv:1909.11646.

[18] Arik, S. O., Chrzanowski, M., Coates, A., Diamos, G., Gibiansky, A., Kang, Y., ... & Shoeybi, M. (2017). Deep voice: Real-time neural text-to-speech. arXiv preprint arXiv:1702.07825.

[19] Gibiansky, A., Arik, S., Diamos, G., Miller, J., Peng, K., Ping, W., ... & Zhou, Y. (2017). Deep voice 2: Multi-speaker neural text-to-speech. In Advances in neural information processing systems (pp. 2962-2970).

[20] Ping, W., Peng, K., Gibiansky, A., Arik, S. O., Kannan, A., Narang, S., ... & Miller, J. (2017). Deep voice 3: Scaling text-to-speech with convolutional sequence learning. arXiv preprint arXiv:1710.07654.

Reference

[21] Wang, Y., Skerry-Ryan, R. J., Stanton, D., Wu, Y., Weiss, R. J., Jaitly, N., ... & Saurous, R. A. (2017). Tacotron: Towards end-to-end speech synthesis. arXiv preprint arXiv:1703.10135.

[22] Shen, J., Pang, R., Weiss, R. J., Schuster, M., Jaitly, N., Yang, Z., ... & Wu, Y. (2018, April). Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4779-4783). IEEE.

[23] Sotelo, J., Mehri, S., Kumar, K., Santos, J. F., Kastner, K., Courville, A., & Bengio, Y. (2017). Char2wav: End-to-end speech synthesis.

[24] Ping, W., Peng, K., & Chen, J. (2018, September). ClariNet: Parallel Wave Generation in End-to-End Text-to-Speech. In International Conference on Learning Representations.

[25] Li, N., Liu, S., Liu, Y., Zhao, S., & Liu, M. (2019, July). Neural speech synthesis with transformer network. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, pp. 6706-6713).

[26] Ren, Y., Ruan, Y., Tan, X., Qin, T., Zhao, S., Zhao, Z., & Liu, T. Y. (2019). Fastspeech: Fast, robust and controllable text to speech. In Advances in Neural Information Processing Systems (pp. 3171-3180).

[27] Ren, Y., Hu, C., Qin, T., Zhao, S., Zhao, Z., & Liu, T. Y. (2020). FastSpeech 2: Fast and High-Quality End-to-End Text-to-Speech. arXiv preprint arXiv:2006.04558.

[28] Donahue, J., Dieleman, S., Bińkowski, M., Elsen, E., & Simonyan, K. (2020). End-to-End Adversarial Text-to-Speech. arXiv preprint arXiv:2006.03575.

[29] Peng, K., Ping, W., Song, Z., & Zhao, K. (2020, November). Non-autoregressive neural text-to-speech. In International Conference on Machine Learning (pp. 7586-7598). PMLR.

[30] Kim, J., Kim, S., Kong, J., & Yoon, S. (2020). Glow-TTS: A Generative Flow for Text-to-Speech via Monotonic Alignment Search. arXiv preprint arXiv:2005.11129.

Reference

[31] Kong, Z., Ping, W., Huang, J., Zhao, K., & Catanzaro, B. (2020). Diffwave: A versatile diffusion model for audio synthesis. arXiv preprint arXiv:2009.09761.

[32] Chen, N., Zhang, Y., Zen, H., Weiss, R. J., Norouzi, M., & Chan, W. (2020). WaveGrad: Estimating gradients for waveform generation. arXiv preprint arXiv:2009.00713.

[33] Valle, R., Shih, K., Prenger, R., & Catanzaro, B. (2020). Flowtron: an Autoregressive Flow-based Generative Network for Text-to-Speech Synthesis. arXiv preprint arXiv:2005.05957.

[34] Taigman, Y., Wolf, L., Polyak, A., & Nachmani, E. (2018, February). VoiceLoop: Voice Fitting and Synthesis via a Phonological Loop. In International Conference on Learning Representations.

[35] Vasquez, S., & Lewis, M. (2019). Melnet: A generative model for audio in the frequency domain. arXiv preprint arXiv:1906.01083.

[36] Kong, J., Kim, J., & Bae, J. (2020). HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis. Advances in Neural Information Processing Systems, 33.

[37] Yu, C., Lu, H., Hu, N., Yu, M., Weng, C., Xu, K., ... & Yu, D. (2019). Durian: Duration informed attention network for multimodal synthesis. arXiv preprint arXiv:1909.01700.

[38] Yang, G., Yang, S., Liu, K., Fang, P., Chen, W., & Xie, L. (2020). Multi-band MelGAN: Faster Waveform Generation for High-Quality Text-to-Speech. arXiv preprint arXiv:2005.05106.

[39] Chorowski, J. K., Bahdanau, D., Serdyuk, D., Cho, K., & Bengio, Y. (2015). Attention-based models for speech recognition. Advances in neural information processing systems, 28, 577-585.

[40] Chiu, C. C., & Raffel, C. (2018, February). Monotonic Chunkwise Attention. In International Conference on Learning Representations.

Reference

[41] Zhang, J. X., Ling, Z. H., & Dai, L. R. (2018, April). Forward attention in sequence-to-sequence acoustic modeling for speech synthesis. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 4789-4793). IEEE.

[42] Tachibana, H., Uenoyama, K., & Aihara, S. (2018, April). Efficiently trainable text-to-speech system based on deep convolutional networks with guided attention. ICASSP 2018.

[43] Chen, M., Tan, X., Ren, Y., Xu, J., Sun, H., Zhao, S., Qin, T. MultiSpeech: Multi-Speaker Text to Speech with Transformer. INTERSPEECH 2020

[44] Gazor, Saeed, and Wei Zhang. "Speech probability distribution." IEEE Signal Processing Letters 10.7 (2003): 204-207.

[45] Usman, Mohammed, et al. "Probabilistic modeling of speech in spectral domain using maximum likelihood estimation." Symmetry 2018

[46] Wang, Y., Stanton, D., Zhang, Y., Ryan, R. S., Battenberg, E., Shor, J., ... & Saurous, R. A. (2018, July). Style Tokens: Unsupervised Style Modeling, Control and Transfer in End-to-End Speech Synthesis. In International Conference on Machine Learning (pp. 5180-5189).

[47] Skerry-Ryan, R. J., Battenberg, E., Xiao, Y., Wang, Y., Stanton, D., Shor, J., ... & Saurous, R. A. (2018, July). Towards End-to-End Prosody Transfer for Expressive Speech Synthesis with Tacotron. In International Conference on Machine Learning (pp. 4693-4702).

[48] Sun, G., Zhang, Y., Weiss, R. J., Cao, Y., Zen, H., Rosenberg, A., ... & Wu, Y. (2020, May). Generating diverse and natural text-to-speech samples using a quantized fine-grained vae and autoregressive prosody prior. ICASSP 2020

[49] Zeng, Z., Wang, J., Cheng, N., & Xiao, J. (2020). Prosody Learning Mechanism for Speech Synthesis System Without Text Length Limit. Proc. Interspeech 2020, 4422-4426.

[50] Sun, G., Zhang, Y., Weiss, R. J., Cao, Y., Zen, H., & Wu, Y. (2020, May). Fully-hierarchical fine-grained prosody modeling for interpretable speech synthesis. ICASSP 2020

Reference

[51] Choi, S., Han, S., Kim, D., & Ha, S. (2020). Attentron: Few-Shot Text-to-Speech Utilizing Attention-Based Variable-Length Embedding. arXiv preprint arXiv:2005.08484.

[52] Chen, M., Tan, X., Li, B., Liu, Y., Qin, T., Zhao, S., & Liu, T. (2020). AdaSpeech: Adaptive Text to Speech for Custom Voice. ICLR 2021

[53] Hayashi, T., Watanabe, S., Toda, T., Takeda, K., Toshniwal, S., & Livescu, K. (2019). Pre-Trained Text Embeddings for Enhanced Text-to-Speech Synthesis}}. Proc. Interspeech 2019, 4430-4434.

[54] Fang, W., Chung, Y. A., & Glass, J. (2019). Towards transfer learning for end-to-end speech synthesis from deep pre-trained language models. arXiv preprint arXiv:1906.07307.

[55] Xiao, Y., He, L., Ming, H., & Soong, F. K. (2020, May). Improving Prosody with Linguistic and Bert Derived Features in Multi-Speaker Based Mandarin Chinese Neural TTS. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 6704-6708). IEEE.

[56] Habib, R., Mariooryad, S., Shannon, M., Battenberg, E., Skerry-Ryan, R. J., Stanton, D., ... & Bagby, T. (2019, September). Semi-Supervised Generative Modeling for Controllable Speech Synthesis. In International Conference on Learning Representations.

[57] Hsu, W. N., Zhang, Y., Weiss, R. J., Zen, H., Wu, Y., Wang, Y., ... & Pang, R. (2018, September). Hierarchical Generative Modeling for Controllable Speech Synthesis. In International Conference on Learning Representations.

[58] Hsu, W. N., Zhang, Y., Weiss, R. J., Chung, Y. A., Wang, Y., Wu, Y., & Glass, J. (2019, May). Disentangling correlated speaker and noise for speech synthesis via data augmentation and adversarial factorization. In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 5901-5905). IEEE.

[59] Zhang, C., Ren, Y., Tan, X., Liu, J., Zhang, K., Qin, T., ... & Liu, T. Y. (2020). Denoising Text to Speech with Frame-Level Noise Modeling. arXiv preprint arXiv:2012.09547.

[60] Ren, Y., Tan, X., Qin, T., Zhao, S., Zhao, Z., & Liu, T. Y. (2019, May). Almost Unsupervised Text to Speech and Automatic Speech Recognition. In International Conference on Machine Learning (pp. 5410-5419).

Reference

[61] Xu, J., Tan, X., Ren, Y., Qin, T., Li, J., Zhao, S., & Liu, T. Y. (2020, August). Lrspeech: Extremely low-resource speech synthesis and recognition. KDD 2020.

[62] Baevski, A., Auli, M., & Mohamed, A. (2019). Effectiveness of self-supervised pre-training for speech recognition. arXiv preprint arXiv:1911.03912.

[63] Chung, Y. A., Wang, Y., Hsu, W. N., Zhang, Y., & Skerry-Ryan, R. J. (2019, May). Semi-supervised training for improving data efficiency in end-to-end speech synthesis. ICASSP 2019

[64] Liu, A. H., Tu, T., Lee, H. Y., & Lee, L. S. (2020, May). Towards unsupervised speech recognition and synthesis with quantized speech representation learning. ICASSP 2020.

[65] Tjandra, A., Sakti, S., & Nakamura, S. (2017, December). Listening while speaking: Speech chain by deep learning. ASRU 2017.

[66] Chen, Y., Assael, Y., Shillingford, B., Budden, D., Reed, S., Zen, H., ... & de Freitas, N. Sample Efficient Adaptive Text-to-Speech. In International Conference on Learning Representations.

[67] Arik, S., Chen, J., Peng, K., Ping, W., & Zhou, Y. (2018). Neural voice cloning with a few samples. In Advances in Neural Information Processing Systems (pp. 10019-10029).

[68] Jia, Y., Zhang, Y., Weiss, R., Wang, Q., Shen, J., Ren, F., ... & Wu, Y. (2018). Transfer learning from speaker verification to multispeaker text-to-speech synthesis. In Advances in neural information processing systems (pp. 4480-4490).

[69] Tu, T., Chen, Y. J., Liu, A. H., & Lee, H. Y. (2020). Semi-Supervised Learning for Multi-Speaker Text-to-Speech Synthesis Using Discrete Speech Representation. Proc. Interspeech 2020, 3191-3195.

[70] Hsu, P. C., & Lee, H. Y. (2020). WG-WaveNet: Real-Time High-Fidelity Speech Synthesis Without GPU. Proc. Interspeech 2020, 210-214.

Reference

[71] Wang, X., Ming, H., He, L., & Soong, F. K. (2020). s-Transformer: Segment-Transformer for Robust Neural Speech Synthesis. arXiv preprint arXiv:2011.08480.

[72] Chen, Y. J., Tu, T., Yeh, C. C., & Lee, H. Y. (2019). End-to-End Text-to-Speech for Low-Resource Languages by Cross-Lingual Transfer Learning}}. Proc. Interspeech 2019, 2075-2079.

[73] Battenberg, E., Skerry-Ryan, R. J., Mariooryad, S., Stanton, D., Kao, D., Shannon, M., & Bagby, T. (2020, May). Location-relative attention mechanisms for robust long-form speech synthesis. ICASSP 2020.