NAACL 2025

👀🌮🤖

Very Attentive Tacotron *(Talk-o-Tron)*

Robust and Unbounded Length Generalization in Autoregressive Transformer-Based Text-to-Speech

Eric Battenberg, RJ Skerry-Ryan, Daisy Stanton, Soroosh Mariooryad, Matt Shannon, Julian Salazar, David Kao

Google DeepMind

Google

Transformer-Based Text-to-Speech

- Autoregressive Transformer-based TTS systems
  - Unparalleled naturalness/quality/expressiveness
  - Can effectively scale up to large datasets.
- However, they have trouble paying attention to the input
  - Poor text adherence (repeat / drop words)
  - Especially with text containing repeated words/digits.
  - Cannot generalize beyond the max training length
- This problem also existed pre-Transformer (in attention-based seq2seq systems)
  - But it's worse with Transformers

Example

Input:

`My phone number is 1-800-9999-2.`

Output:

`My phone number is 1-800-99999999-2.`

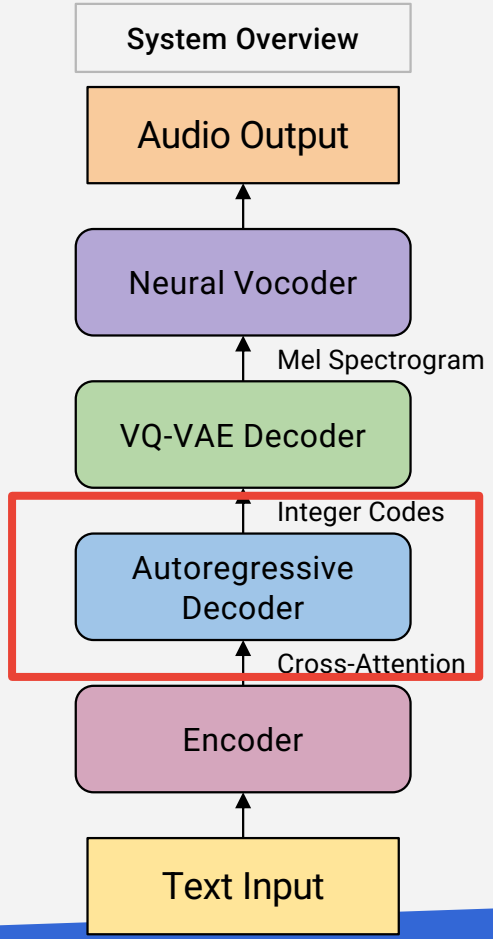
Existing Approaches to Improving Robustness

- Go back to doing explicit duration modeling using force aligned training data *[ELLA-V]*
- Use an RNN-T-like Transducer mechanism *[VALL-T]*
- Limit cross-attention to a single layer with a single head over a narrow monotonically advancing window *[MQ-TTS]*

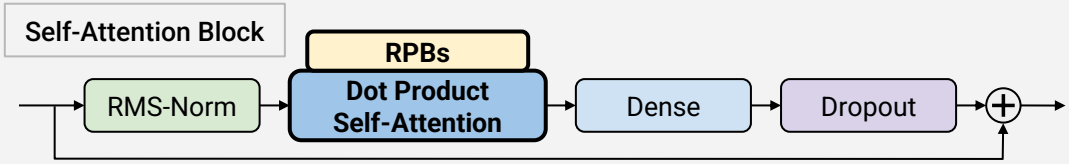
- *These complicate training, hurt inference efficiency, and/or reduce the power of the standard Transformer architecture.*

Basic Discrete TTS System

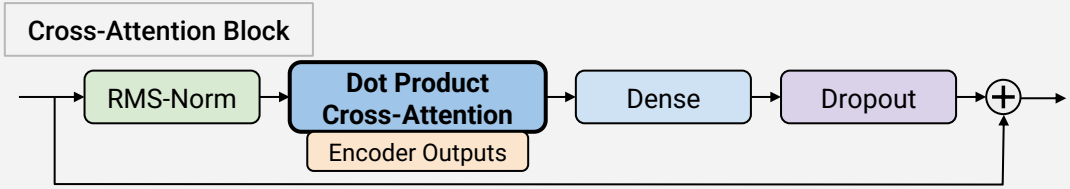
- Encoder-Decoder model
- Text (phonemes) processed by bidirectional encoder
  - Encoder = conv + self-attention
- Transformer-based decoder
  - Attends to encoded text using multiple cross-attention layers
  - Models sequence of integer codes produced by VQ-VAE
- VQ-VAE converts integer codes to mel spectrogram
- Neural vocoder converts mel spectrogram to waveform



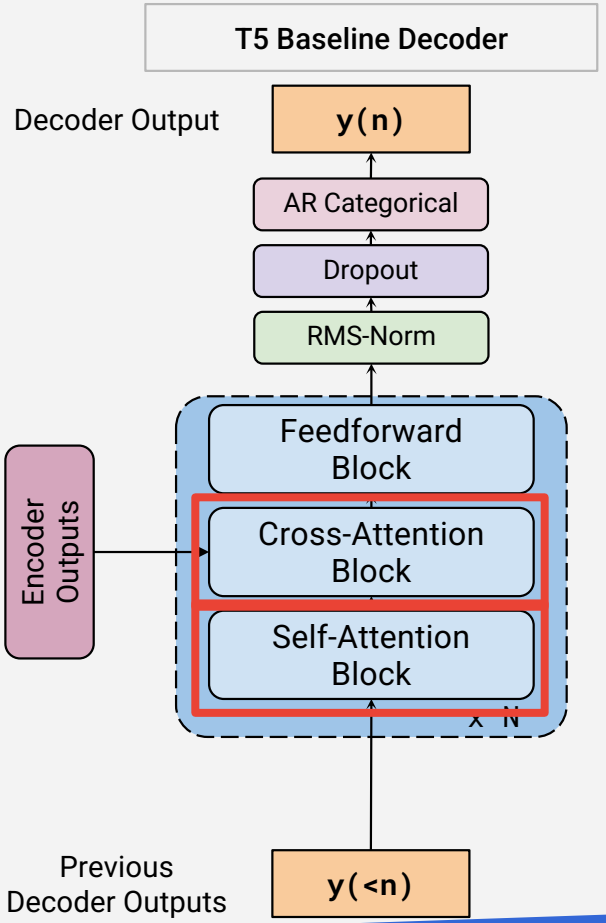
T5 Baseline Decoder



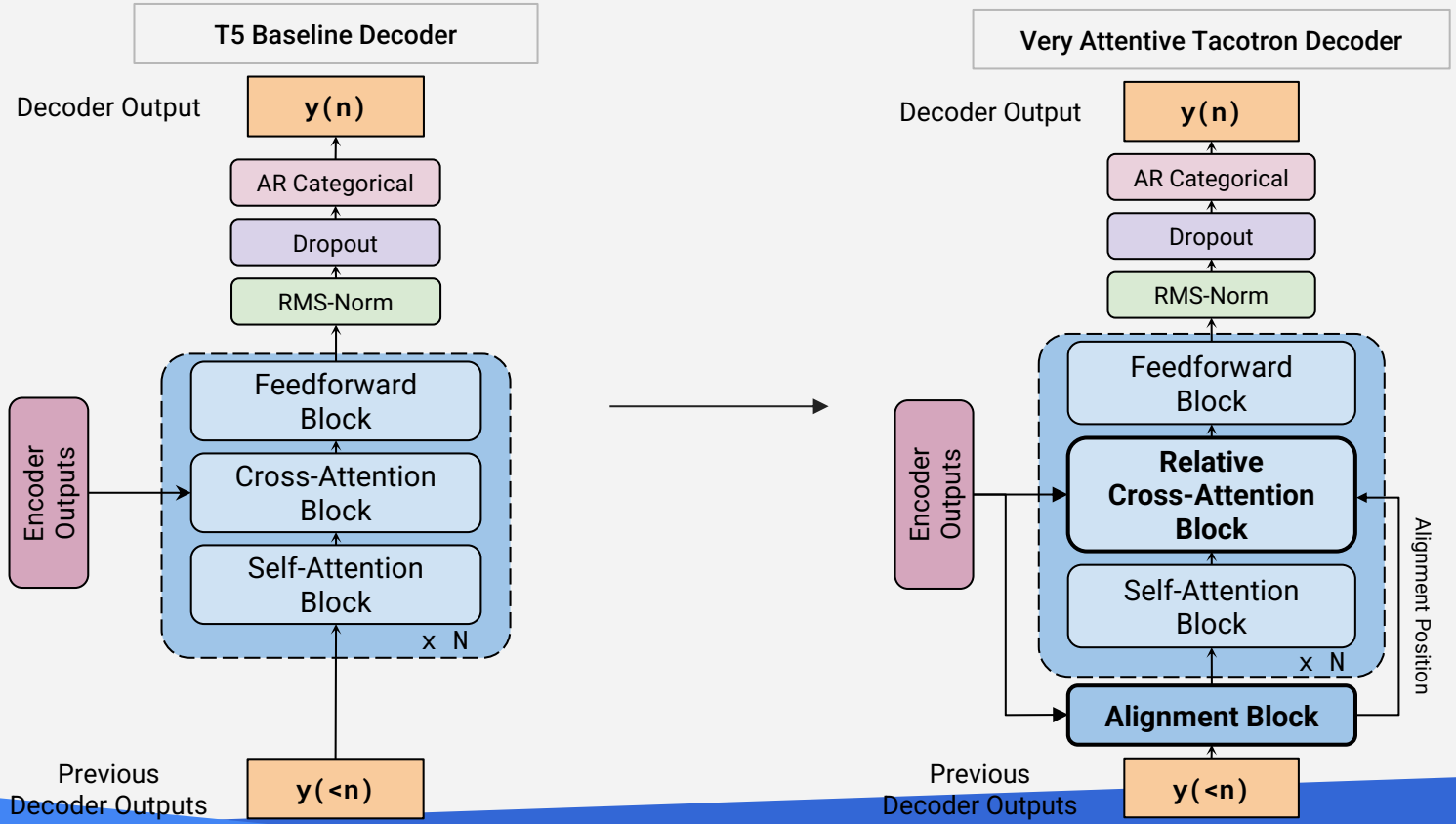
- RPBs = Relative Position Biases
  - Matrix of attention biases for each relative position



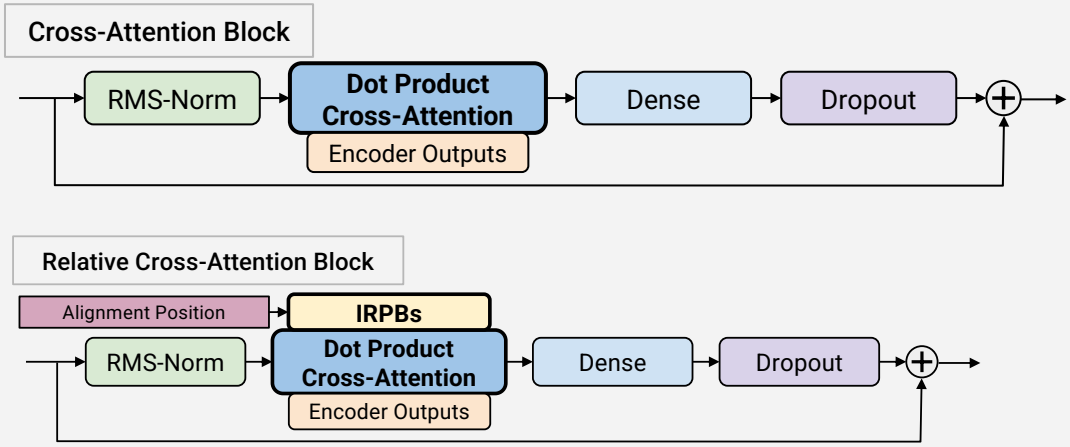
- Problem: No cross-attention RPBs
  - No sense of relative position between input/output
  - Leads to stability problems



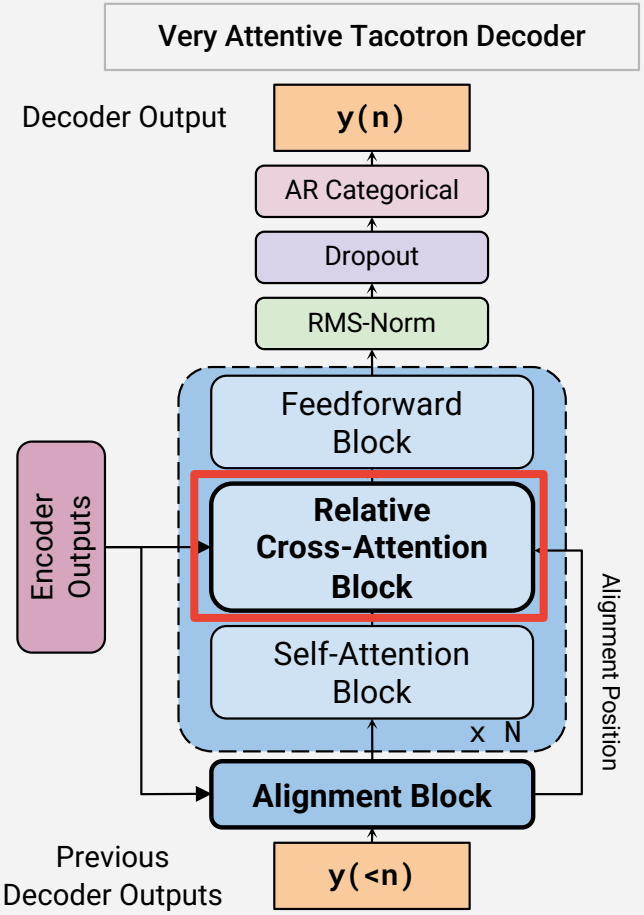
T5 Baseline Decoder -> Very Attentive Decoder



Very Attentive Decoder

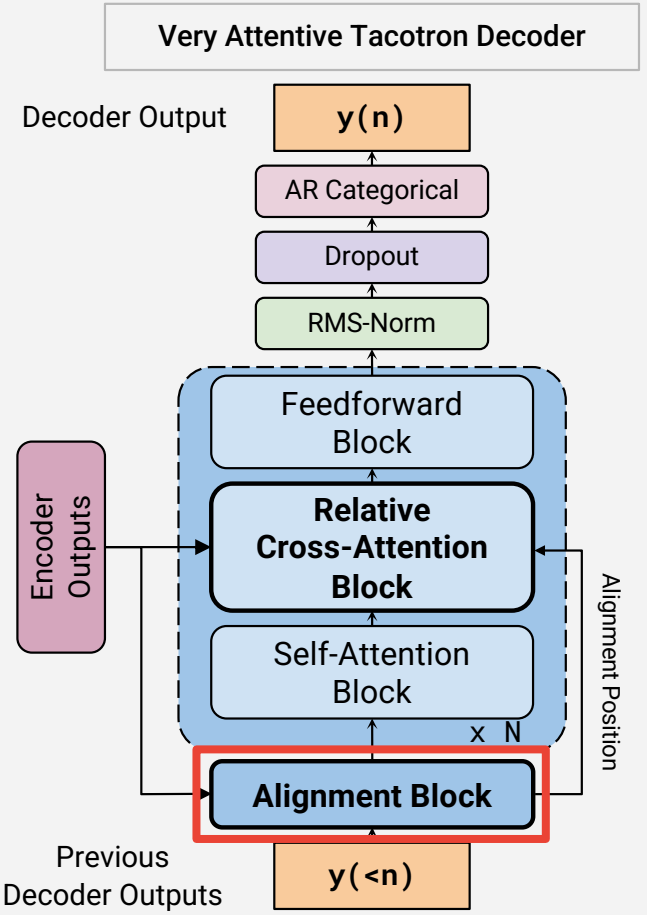


- Cross-attention relative positions computed using alignment position as “zero” position.
  - Non-integer valued, differentiable.
- RPBs are only defined for integer relative positions.
- IRPBs = Interpolated Relative Position Biases
  - Linearly interpolate between adjacent RPB values



Very Attentive Tacotron Decoder

- Alignment block produces monotonic alignment position
  - Position is latent and cannot be teacher forced
  - Updated each decoder step by small RNN-based subnetwork
- Alignment position provides relative positions to IRPBs in all cross-attention layers



Experiments

- Models trained on private and public (LibriTTS) datasets:
  - T5 Baseline
  - Very Attentive Tacotron (VAT)
  - Additional baselines: Tacotron-GMMA, Non-Attentive Tacotron (NAT)
- Evaluations:
  - Naturalness (MOS / SxS)
  - Robustness (ASR-based CER)
  - Length generalization (ASR-based CER)
  - Repeated words stress test

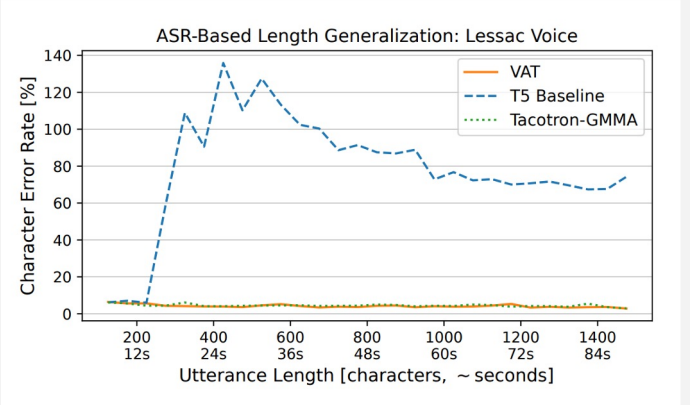
Results: Naturalness, Robustness

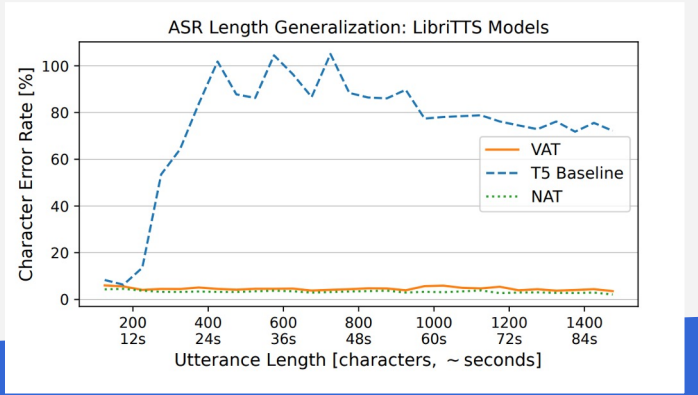
- VAT matches naturalness of T5 baseline (MOS/SxS)
  - Beats Tacotron-GMMA / NAT
- VAT has superior robustness compared to T5 baseline (CER).

Lessac Voice	MOS	SxS vs VAT	CER
Ground Truth	4.00 ±0.07		2.9
VAT	3.68 ±0.08	—	3.3
T5 Baseline	3.75 ±0.07	-0.06 ±0.14	10.2
Tacotron-GMMA[3]	3.62 ±0.08	**-0.32 ±0.14**	3.7
LibriTTS	MOS	SxS vs VAT	CER
Ground Truth	3.70 ±0.09		3.6
VAT	3.16 ±0.09	—	4.6
T5 Baseline	3.07 ±0.09	0.01 ±0.14	10.7
NAT	3.22 ±0.08	-0.12 ±0.15	3.3

Results: Length Generalization, Repeated Words

- Length generalization:
  - Measure CER for utterances of increasing length.
  - Models trained on 10 sec utterances.
  - T5 baseline fails beyond 12 sec.
  - VAT stable out to 90 sec and beyond.
- Repeated words:
  - Stress test templates
    - 3 templates, 1-9 repetitions each.
    - e.g., "My phone number is 1-800-[9,...,9]-2
  - T5 baseline: Errors on 52% of phrases.
  - VAT: No mistakes.





Repeated Digits Examples

- **T5 Baseline** 
- Input: “My phone number is 1, 800, 9, 9, 9, 9, 2”
- Actual: “My phone number is 1, 800, 9, 9, 9, 9, 9, 9, 9, 9, 2”

- **Very Attentive Tacotron** 
- Input: “My phone number is 1, 800, 9, 9, 9, 9, 9, 9, 9, 9, 9, 2”
- Actual: “My phone number is 1, 800, 9, 9, 9, 9, 9, 9, 9, 9, 9, 2”

Longform Examples

(Models trained on <10sec utterances)

- **T5 Baseline**

- **Very Attentive Tacotron**

Today, we're introducing "Very Attentive Tacotron." It's a new text-to-speech system from Google powered by discrete audio tokens and an autoregressive Transformer. But that's not all! It can faithfully synthesize long text into speech; well over a minute. Yes, that's many times longer than the examples seen during training. No more dropping words, no more repetitions, and no more erratic outputs... all without a duration model!

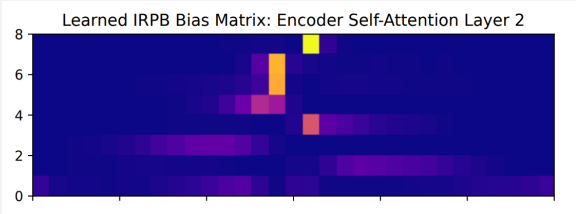
Our approach uses an alignment mechanism to provide cross-attention operations with relative location information. The associated alignment position is learned as a latent property of the model via backprop and requires no external alignment information during training.

While the approach is tailored to the monotonic nature of TTS input-output alignment, it is still able to benefit from the flexible modeling power of interleaved, multi-head, self- and cross-attention operations.

Our system matches the naturalness and expressiveness of a baseline, discrete, autoregressive T5-based system while generalizing to any practical utterance length.

Discussion

- Slight efficiency hit during training due to serialized alignment computations.
  - But not a major issue. Mitigations discussed in paper.
- In the paper we also discuss:
  - Initializing and constraining IRPBs for consistency and stability.
  - Details behind RPBs, IRPBs, relative cross-attention, alignment layer
  - What patterns emerge in learned IRPBs?
  - Training and architecture specifics
  - And much more!
- Future work
  - Apply to other monotonic seq2seq tasks like ASR.
  - Adapting VAT for decoder-only models
    - (They're so hot right now!)



Resources

- Paper:
  - https://arxiv.org/abs/2410.22179
- Audio examples:
  - https://google.github.io/tacotron/publications/very_attentive_tacotron/index.html
- Code examples:
  - https://github.com/google/sequence-layers/blob/main/examples/very_attentive_tacotron.py
  - (Links available in the paper)

References

- [VALL-E] Wang et al. Neural Codec Language Models are Zero-Shot Text to Speech Synthesizers. https://arxiv.org/abs/2301.02111
- [ELLA-V] Song et al. ELLA-V: Stable Neural Codec Language Modeling with Alignment-guided Sequence Reordering. https://arxiv.org/abs/2401.07333
- [VALL-T] Du et al. VALL-T: Decoder-Only Generative Transducer for Robust and Decoding-Controllable Text-to-Speech. https://arxiv.org/abs/2401.14321v4
- [MQ-TTS] Chen et al. A Vector Quantized Approach for Text to Speech Synthesis on Real-World Spontaneous Speech. https://arxiv.org/abs/2302.04215

Very Attentive Tacotron

Robust and Unbounded Length Generalization in Autoregressive Transformer-Based Text-to-Speech

Eric Battenberg, RJ Skerry-Ryan, Daisy Stanton, Soroosh Mariooryad,
Matt Shannon, Julian Salazar, David Kao

Google DeepMind

Thank you!

Google

The End