Speech Generation WaveNet & Tacotron

SAPOZHNIKOVA DARIA

BAMI 181

Approaches of TTS

CONCATENATIVE TTS

High-quality audio clips recordings, which are combined together to form the speech

Pros

- High quality of audio in terms of intelligibility;
- Possibility to preserve the original actor’s voice

Cons

- Such systems are very time consuming
- The resulting speech may sound less natural and emotionless

PARAMETRIC TTS

If we can make approximations of the parameters that make the speech, we can train a model to generate all kinds of speech.

Pros:

- Increased naturalness of the audio.
- Flexibility
- Lower development cost

Cons:

- Lower audio quality in terms of intelligibility
- The voice can sound robotic

WaveNet

Mathematics for waveforms

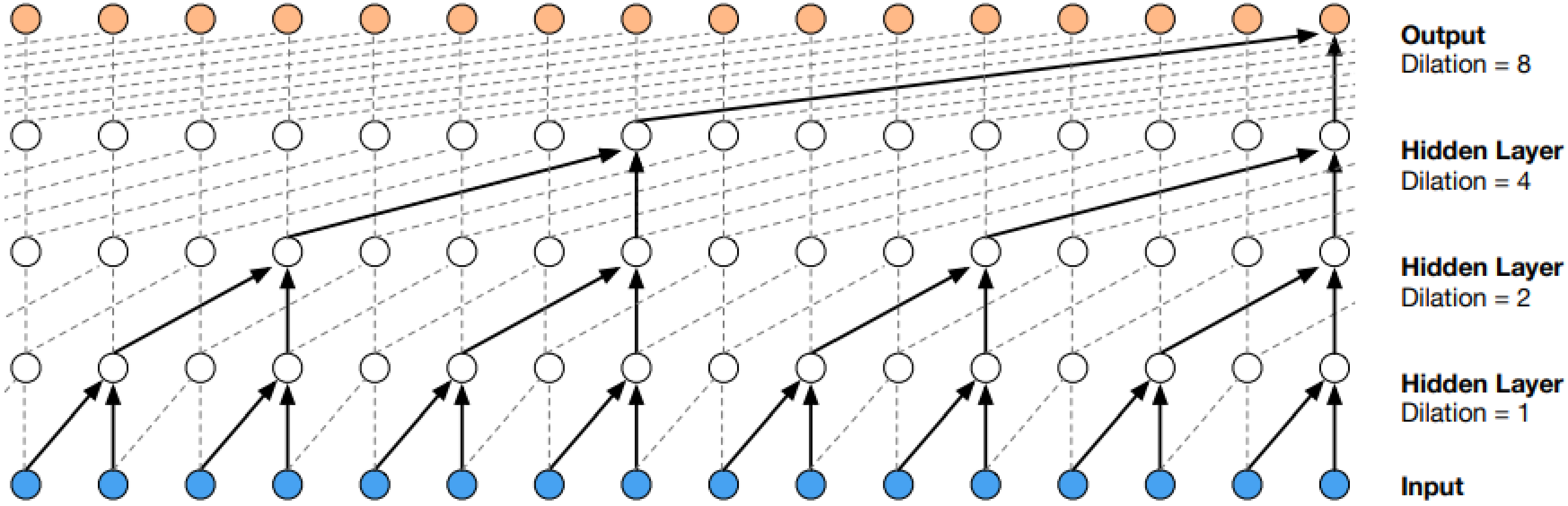
The joint probability of a waveform $x = \{x_1, \ldots, x_T\}$ is factorised as a product of conditional probabilities as follows:

$$p(\mathbf{x}) = \prod_{t=1}^{T} p(x_t \mid x_1, \ldots, x_{t-1})$$

Each audio sample x_t is therefore conditioned on the samples at all previous timesteps

Dilated causal convolutional layers

,



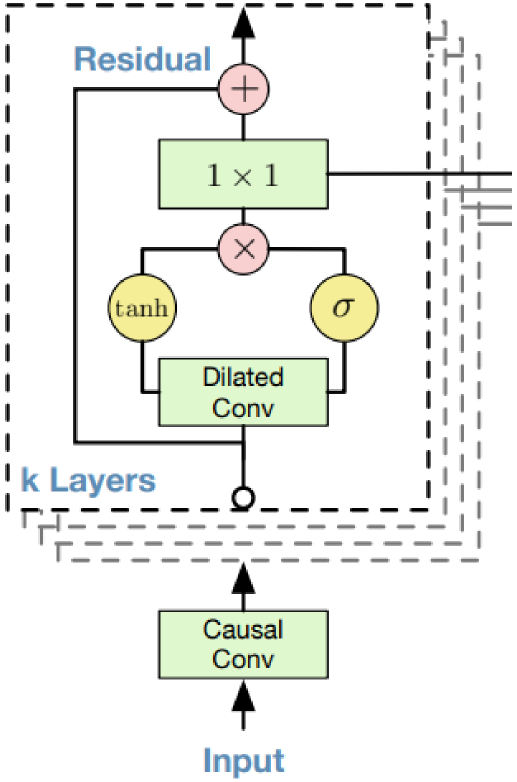
Mathematics for waveforms

Because raw audio is typically stored as a sequence of 16-bit integer values, a softmax layer would need to output 65,536 probabilities per timestep to model all possible values. To make this more tractable, we first apply a μ-law companding to the data, and then quantize it to 256 possible values:

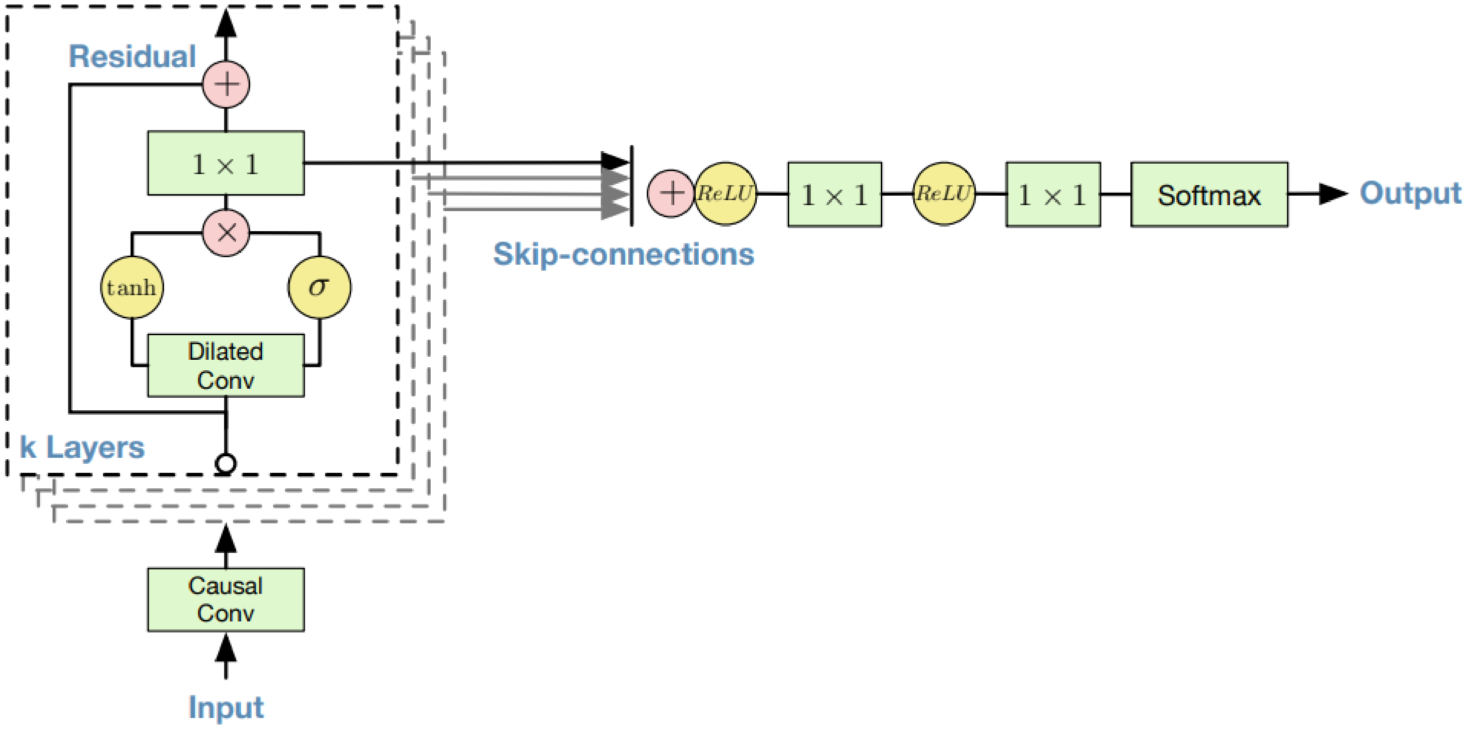
$$f(x_t) = \operatorname{sign}(x_t)\frac{\ln(1 + \mu |x_t|)}{\ln(1 + \mu)}$$

where −1 < x_t < 1 and μ = 255

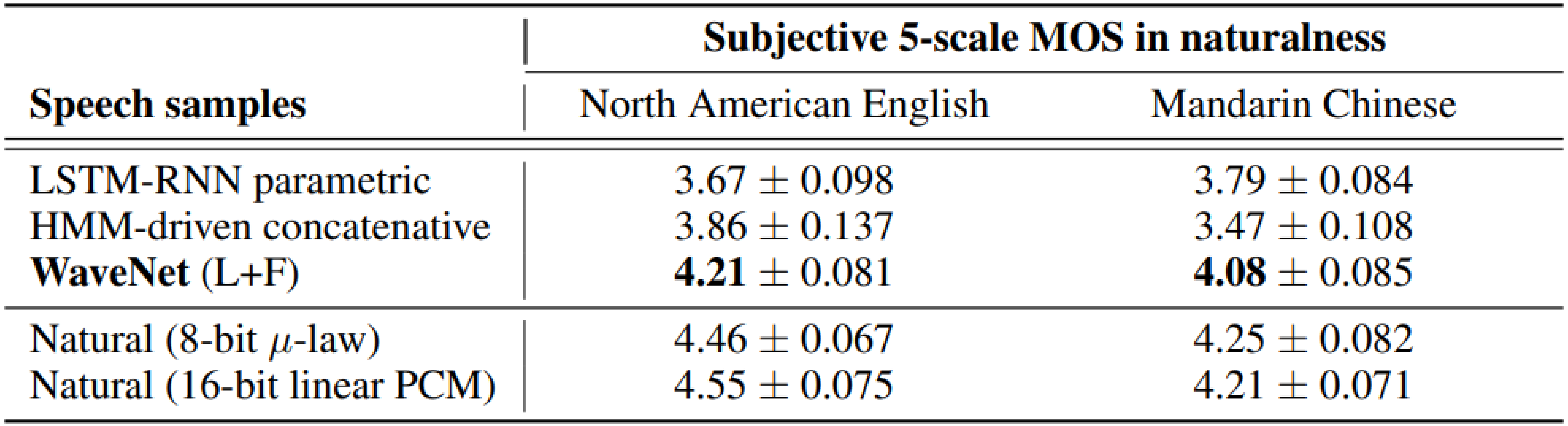
Model architecture



Model architecture



Results

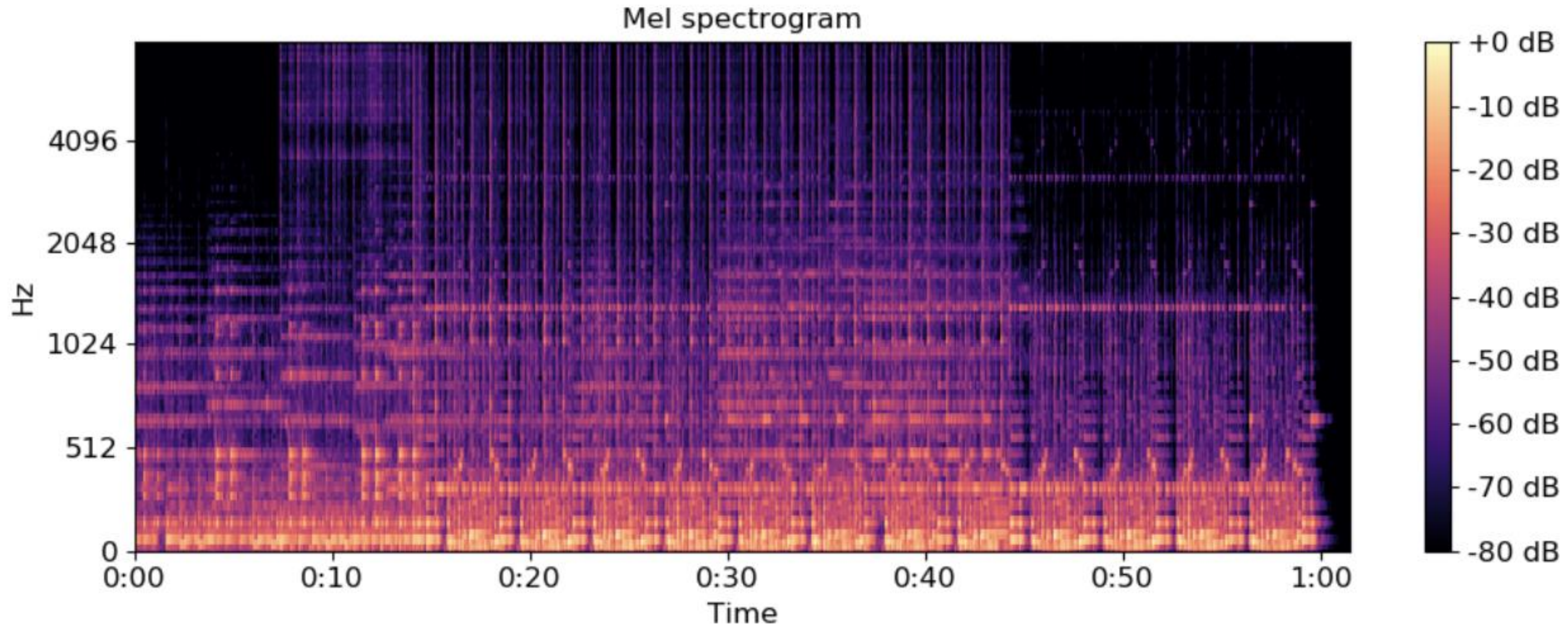


	Subjective 5-scale MOS in naturalness	
Speech samples	North American English	Mandarin Chinese
LSTM-RNN parametric	3.67 ± 0.098	3.79 ± 0.084
HMM-driven concatenative	3.86 ± 0.137	3.47 ± 0.108
WaveNet (L+F)	**4.21** ± 0.081	**4.08** ± 0.085
Natural (8-bit μ-law)	4.46 ± 0.067	4.25 ± 0.082
Natural (16-bit linear PCM)	4.55 ± 0.075	4.21 ± 0.071

Results

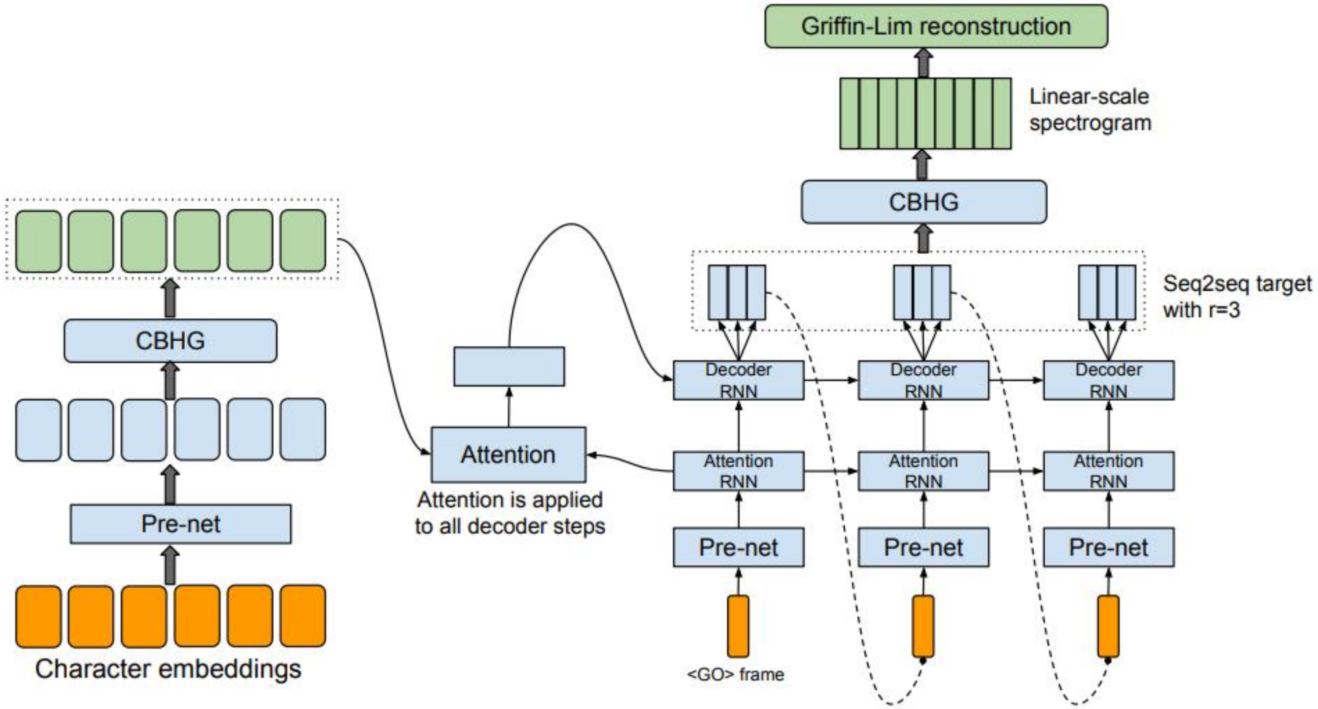
Tacotron

MEL-SPECTOGRAM



Model architecture

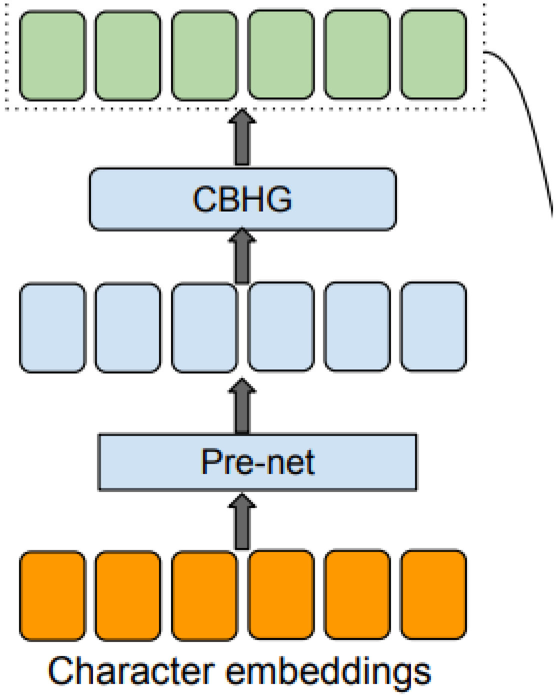
The model takes characters as input and outputs the corresponding raw spectrogram, which is then fed to the Griffin-Lim reconstruction algorithm to synthesize speech



ENCODER

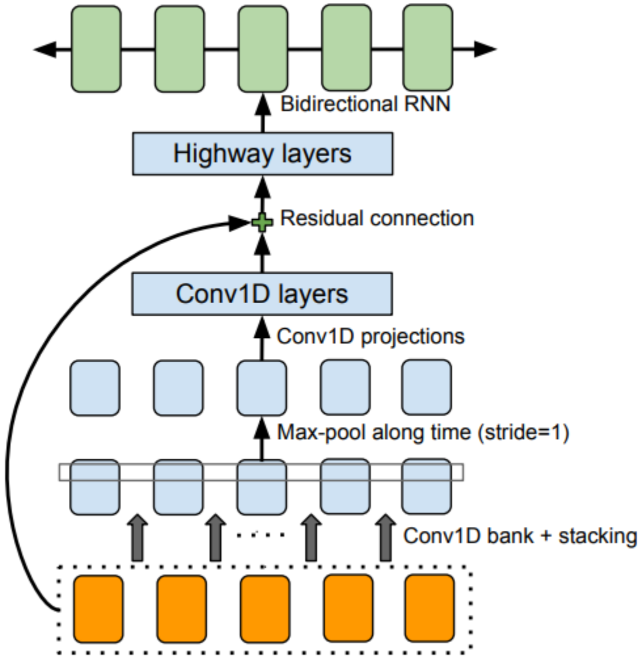
The goal of the encoder is to extract robust sequential representations of text.

- Applying a set of non-linear transformations (pre-net) , to each embedding. In the paper a bottleneck layer with dropout is used as the pre-net, which helps convergence and improves generalization
- A CBHG module transforms the pre-net outputs into the final encoder representation used by the attention module



CBHG Module

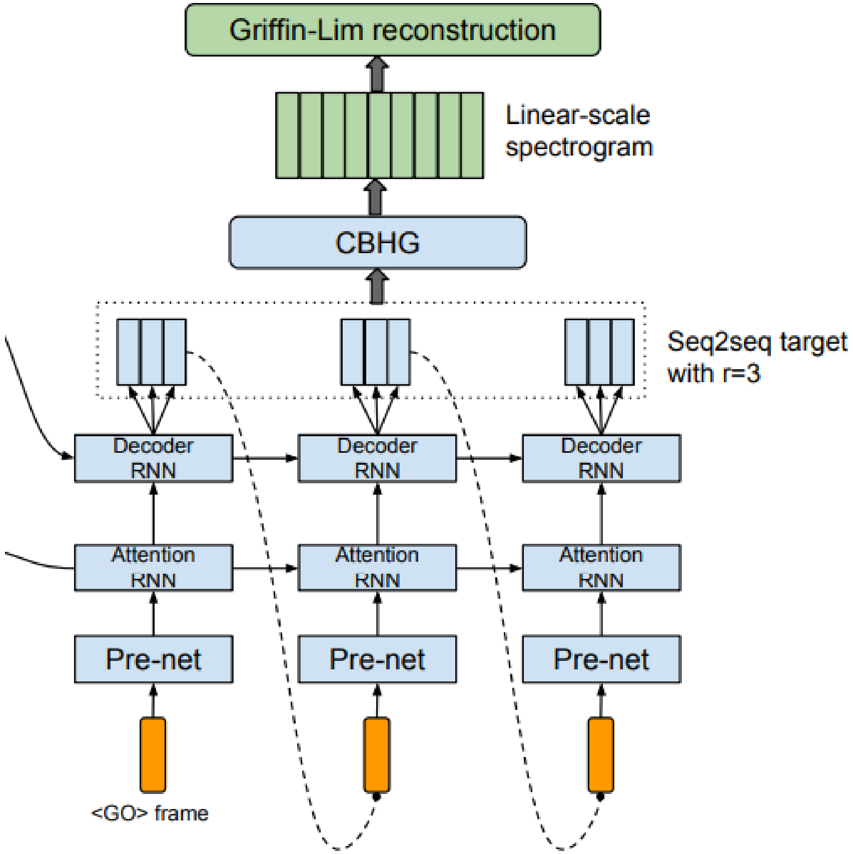
- K sets of 1-D convolution bank with filters, where the k-th set contains Ck filters of width k
- Max-pool along time with the stride of 1 to preserve the original time resolution
- Fixed-width 1-D convolutions, whose outputs are added with the original input sequence via residual connections
- The outputs are fed into a multi-layer highway network to extract high-level features.
- Bidirectional GRU RNN is stacked on top to extract sequential features from both forward and backward context.



DECODER

The decoder target is a 80-band mel-scale spectrogram instead of raw spectrogram

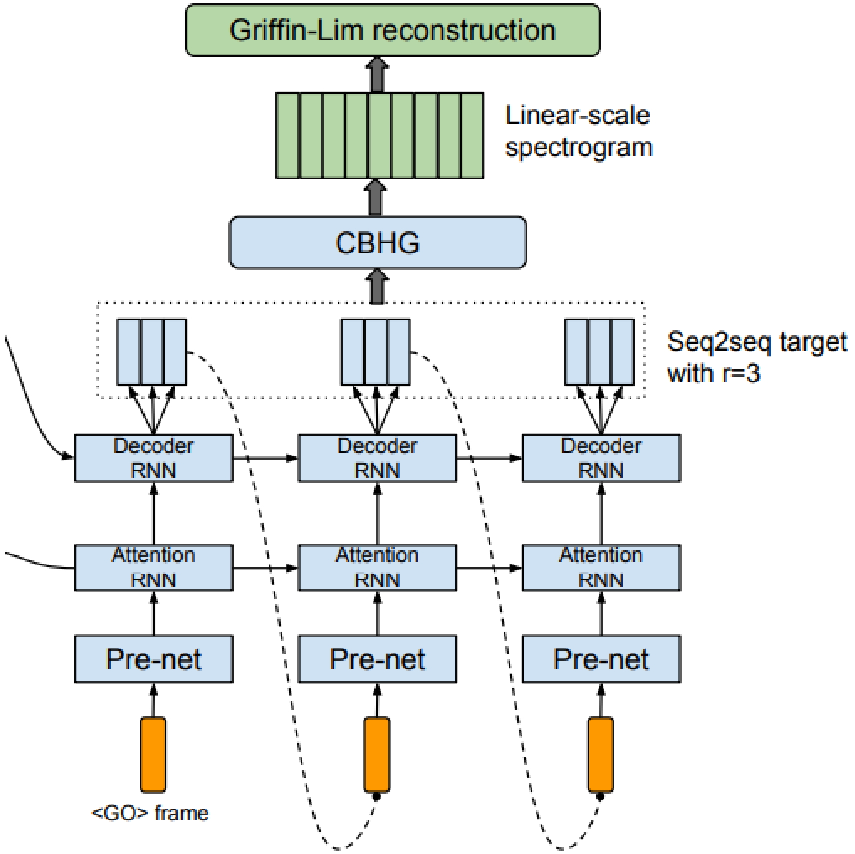
- Concatenate the context vector and the attention RNN cell output to form the input to the decoder RNNs.
- We use a stack of GRUs with vertical residual connections for the decoder.



DECODER

At decoder step t, the last frame of the r predictions is fed as input to the decoder at step t + 1. During training, we always feed every r-th ground truth frame to the decoder.

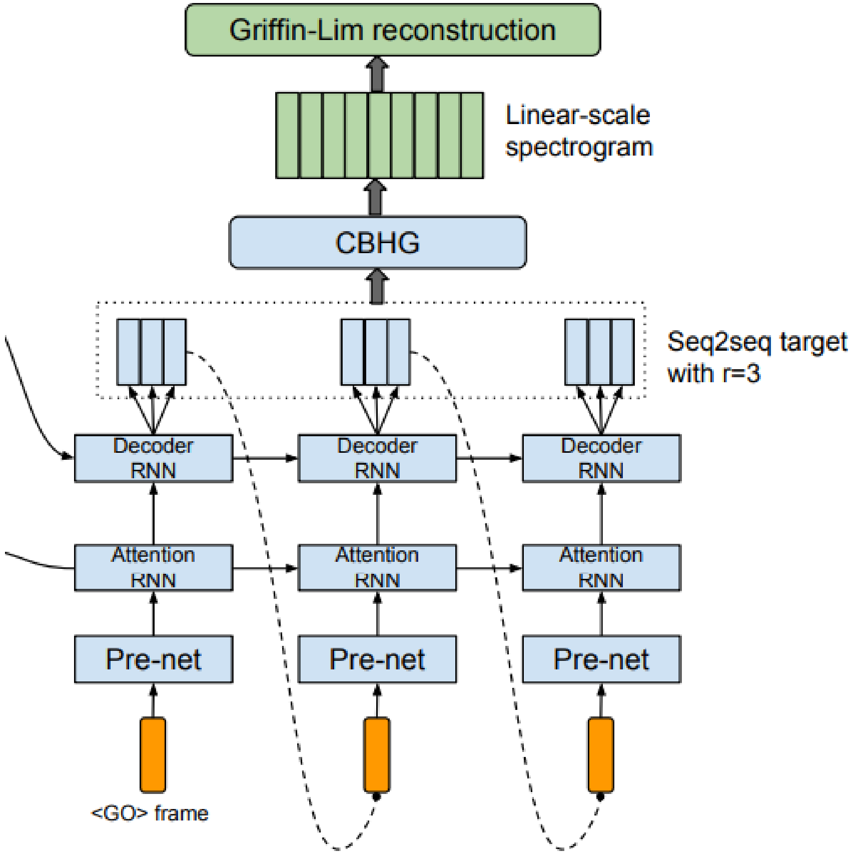
The input frame is passed to a pre-net as is done in the encoder.



POST-PROCESSING NET

The post-processing net's task is to convert the seq2seq target to a target that can be synthesized into waveforms. To be more precise, the post-processing net learns to predict spectral magnitude sampled on a linear-frequency scale.

Another motivation of the post-processing net is that it can see the full decoded sequence. It has both forward and backward information to correct the prediction error for each individual frame.



Results

Database - internal North American English dataset

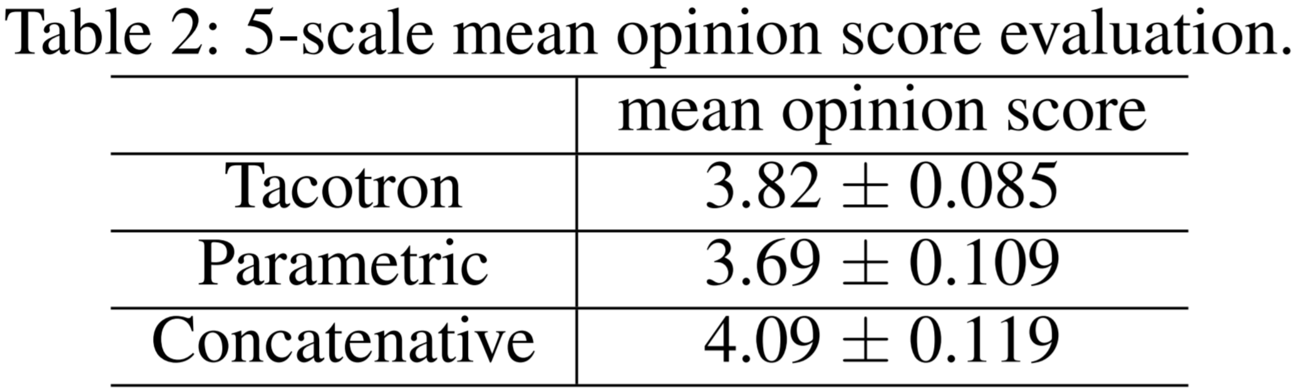
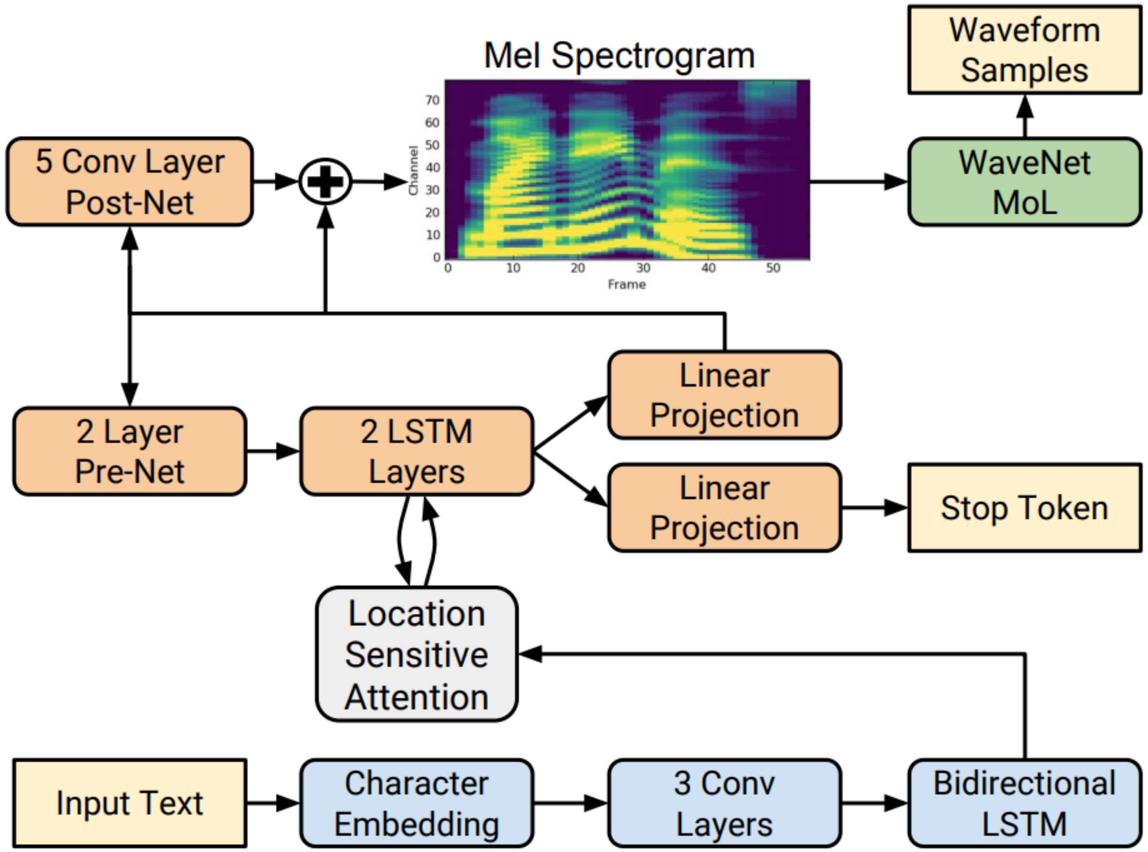


Table 2: 5-scale mean opinion score evaluation.

	mean opinion score
Tacotron	3.82 ± 0.085
Parametric	3.69 ± 0.109
Concatenative	4.09 ± 0.119

Tacotron 2

Model architecture

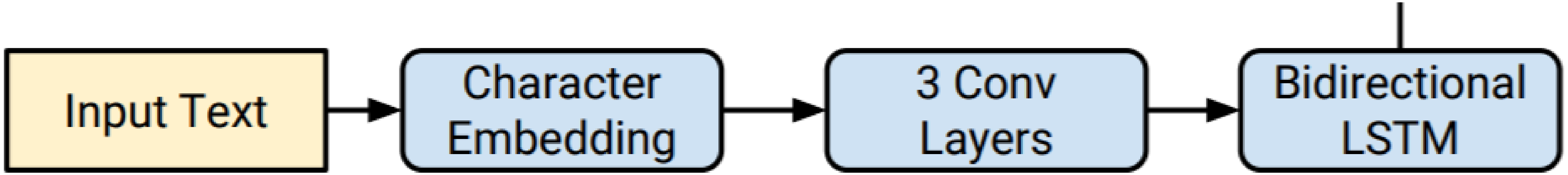


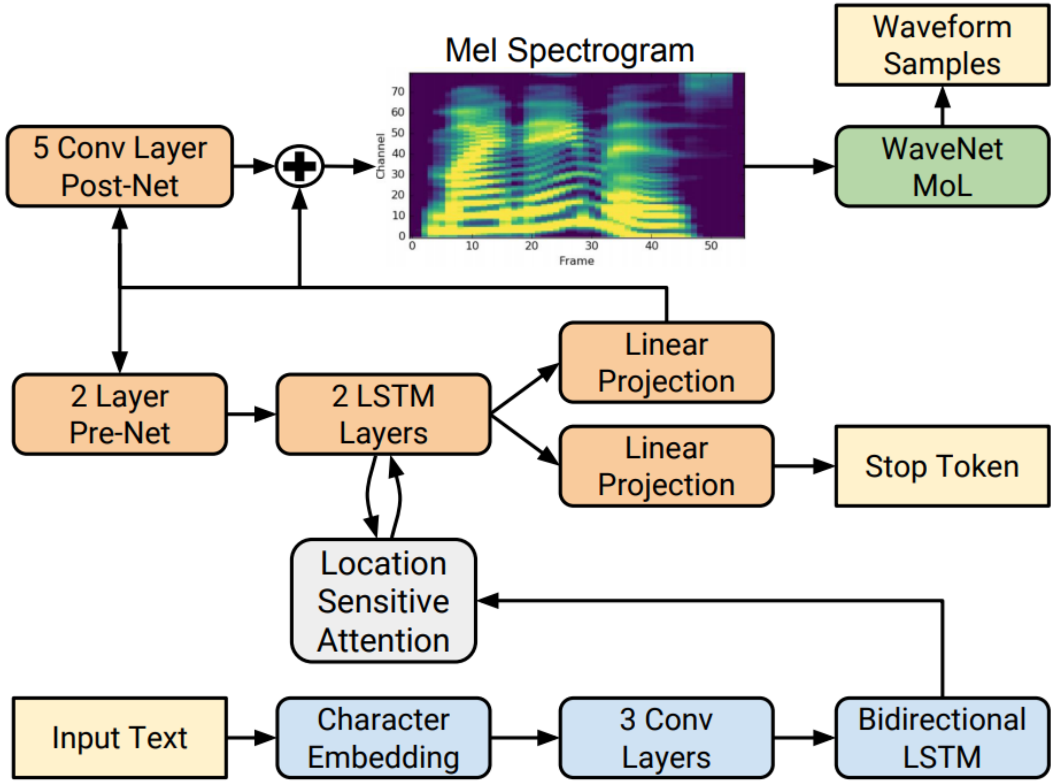
ENCODER

Input characters are represented using a learned 512-dimensional character embedding

The output of the final convolutional layer is passed into a single bi-directional LSTM layer containing 512 units to generate the encoded features

The encoder output is consumed by an attention network which summarizes the full encoded sequence as a fixed-length context vector for each decoder output step.





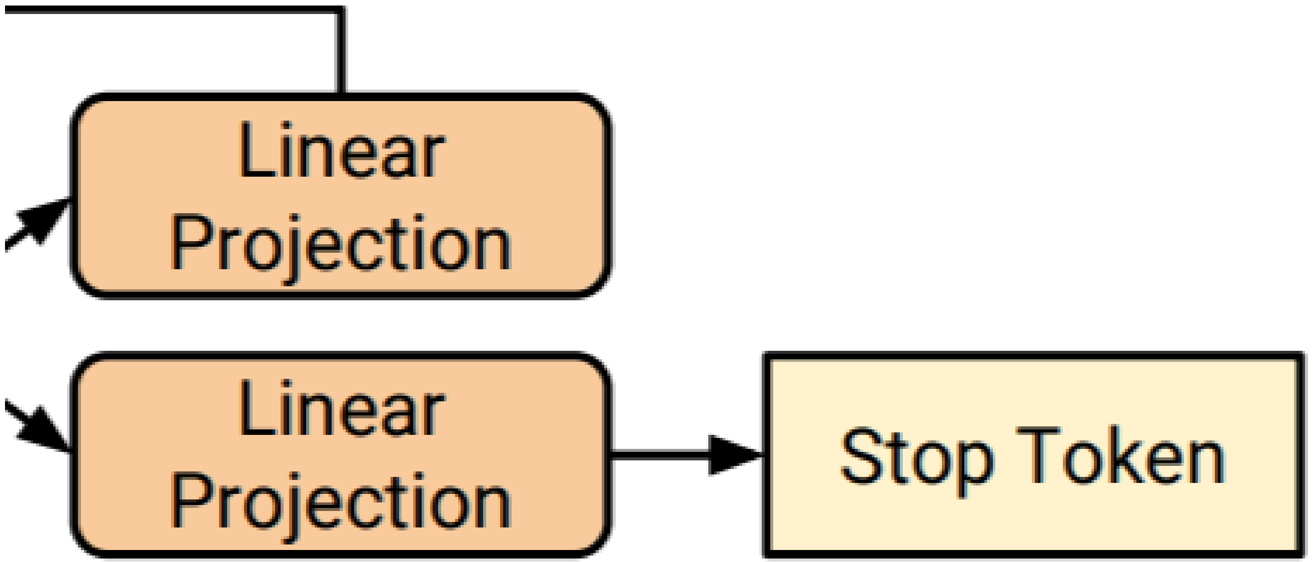
DECODER

The prediction from the previous timestep is first passed through a small pre-net containing 2 fully connected layers of 256 hidden ReLU units.

The pre-net output and attention context vector are concatenated and passed through a stack of 2 uni-directional LSTM layers with 1024 units.

The concatenation of the LSTM output and the attention context vector is projected through a linear transform to predict the target spectrogram frame.

Finally, the predicted mel-spectrogram is passed through a 5-layer convolutional post-net which predicts a residual to add to the prediction to improve the overall reconstruction.



STOP TOKEN

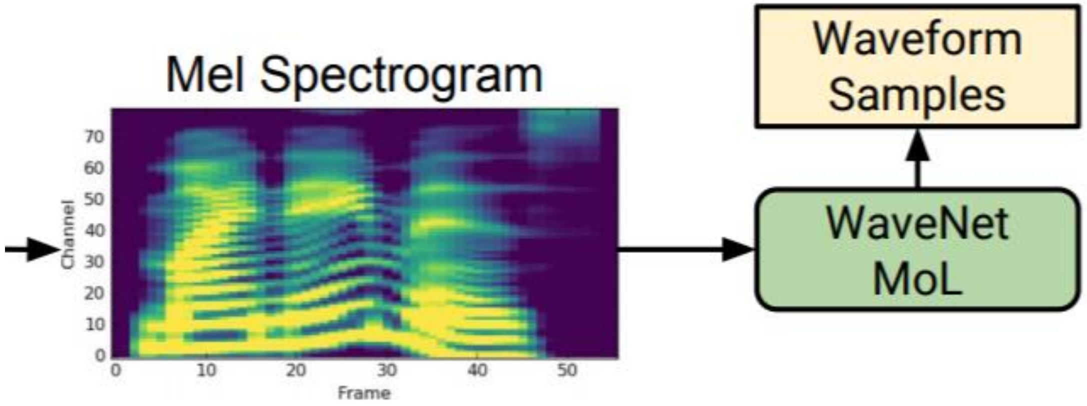
In parallel to spectrogram frame prediction, the concatenation of decoder LSTM output and the attention context is projected down to a scalar and passed through a sigmoid activation to predict the probability that the output sequence has completed.

This “stop token” prediction is used during inference to allow the model to dynamically determine when to terminate generation instead of always generating for a fixed duration.

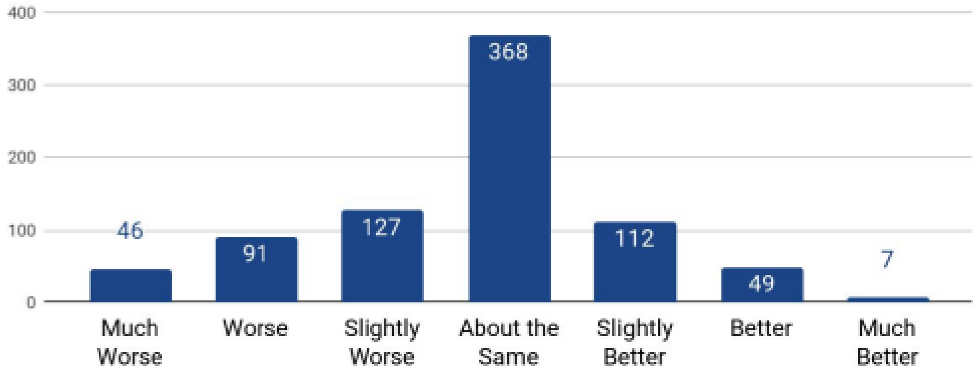
WAVENET

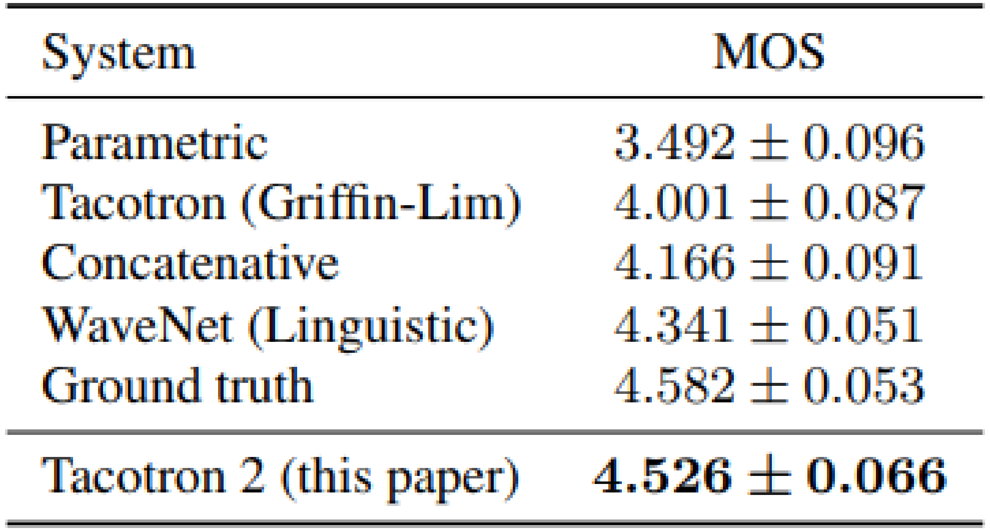
Instead of predicting discretized buckets with a softmax layer, we use a 10- component mixture of logistic distributions (MoL) to generate 16-bit samples at 24 kHz.

To compute the logistic mixture distribution, the WaveNet stack output is passed through a ReLU activation followed by a linear projection to predict parameters



Results





System	MOS
Parametric	3.492 ± 0.096
Tacotron (Griffin-Lim)	4.001 ± 0.087
Concatenative	4.166 ± 0.091
WaveNet (Linguistic)	4.341 ± 0.051
Ground truth	4.582 ± 0.053
Tacotron 2 (this paper)	**4.526 ± 0.066**

Results

In the following examples, one is generated by Tacotron 2, and one is the recording of a human, but which is which?

References. WaveNet

- https://arxiv.org/pdf/1609.03499.pdf
- https://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html
- http://www.machinelearning.ru/wiki/images/3/31/Digital_Signal_Processing%2C_lecture_11.pdf
- https://deepmind.com/blog/article/wavenet-generative-model-raw-audio
- https://habr.com/ru/company/Voximplant/blog/309648/
- https://www.youtube.com/watch?v=YyUXG-BfDbE

References. Tacotron

- https://arxiv.org/pdf/1703.10135.pdf
- https://arxiv.org/pdf/1712.05884.pdf
- https://habr.com/ru/post/465941/
- https://habr.com/ru/post/409257/
- https://www.youtube.com/watch?v=tHAdlv7ThjA
- https://google.github.io/tacotron/index.html