Generating Lo-Fi Music

Ashwin Swar and Jason Xu

1. Project Description
2. Failed Architectures
3. ST-FFT Encoder Decoder Architecture
4. TCN Architecture

Project Description

Goal

- Given a few seconds of input, can we generate the rest of the Lo-Fi song?

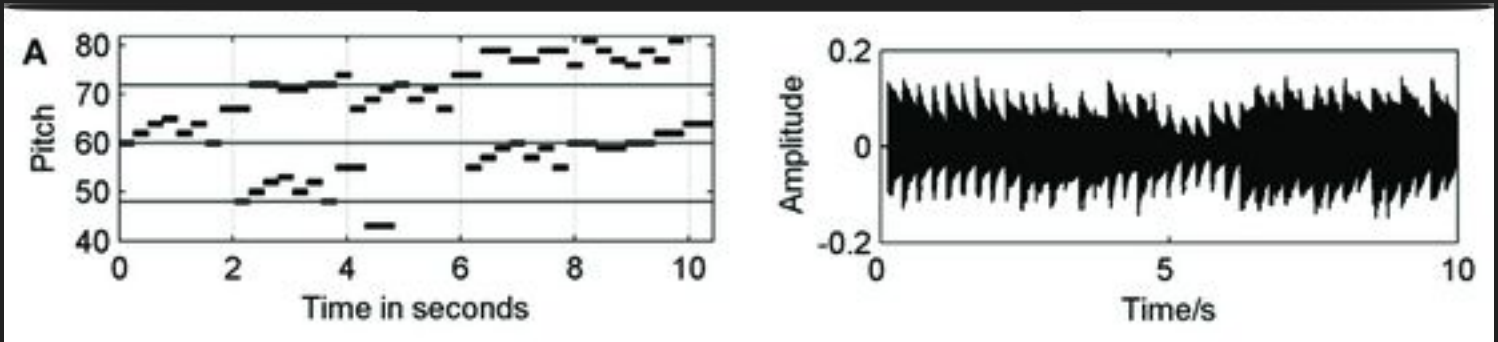
Project Description - Waveform vs MIDI

MIDI:

- Low resolution representation
- Lacks diversity
- Easier to model
- Very popular format for music generation

Waveform:

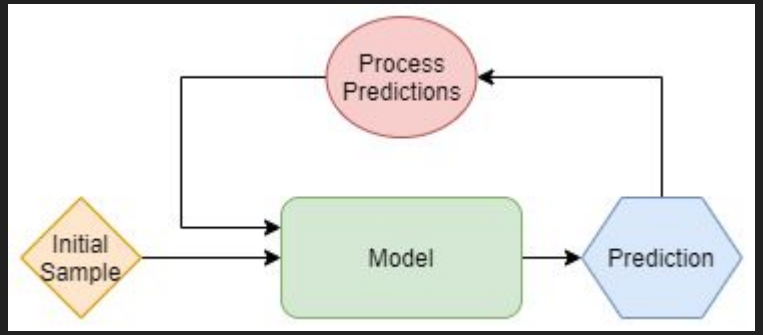
- High resolution representation
- Rich in diversity
- Difficult to model
- Less popular



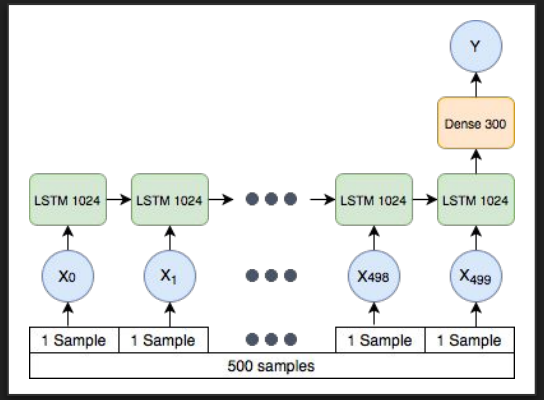
Project Description

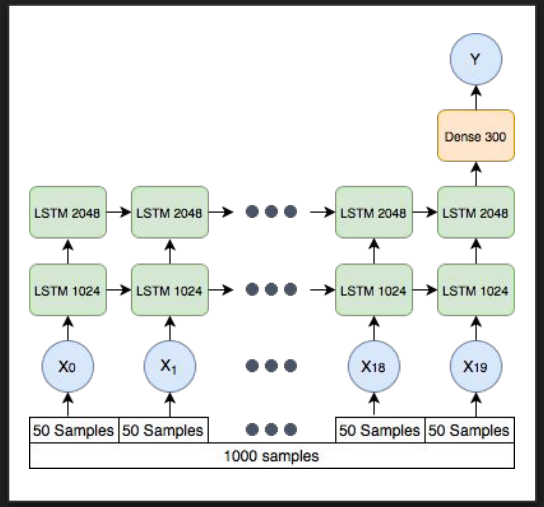
How do we do music generation?

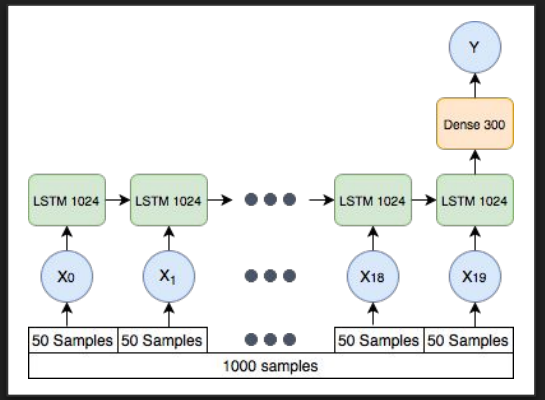
- Shakespeare text example



Failed Architectures



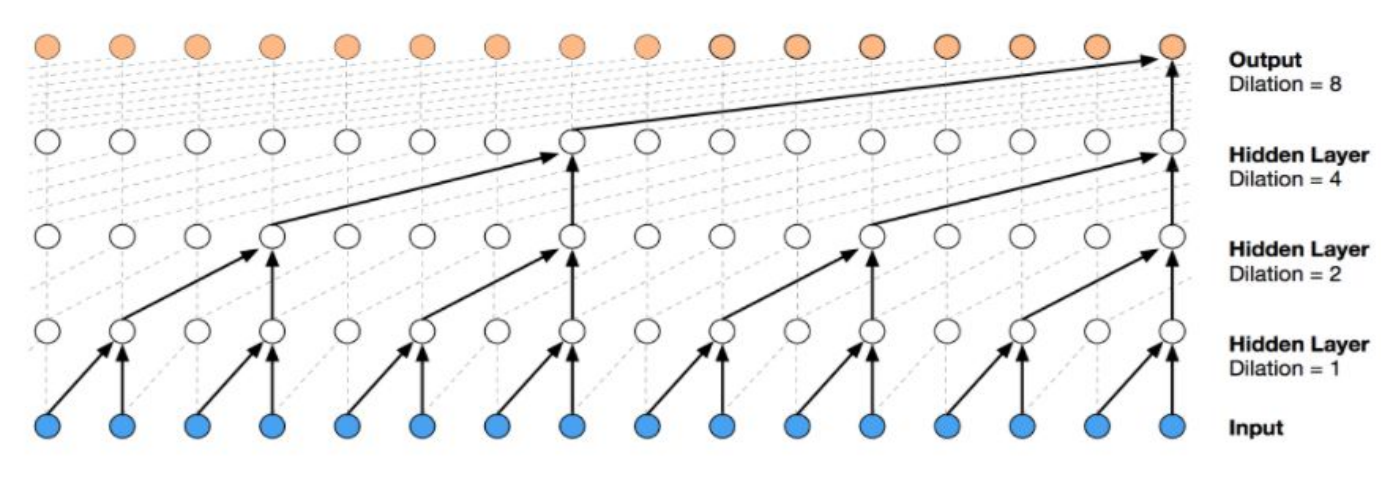




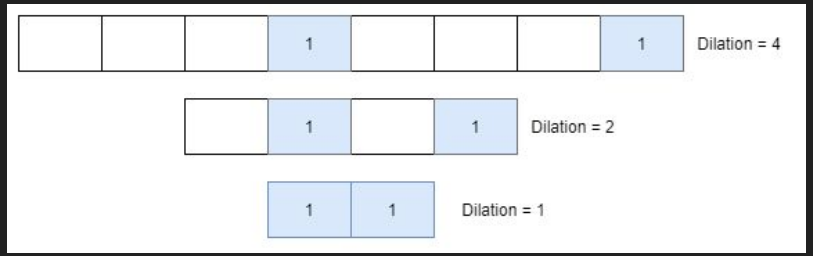
Issues:

Backpropagation through time was too slow, not a large enough sliding window. Constant prediction

Temporal Convolutional Networks(TCN)



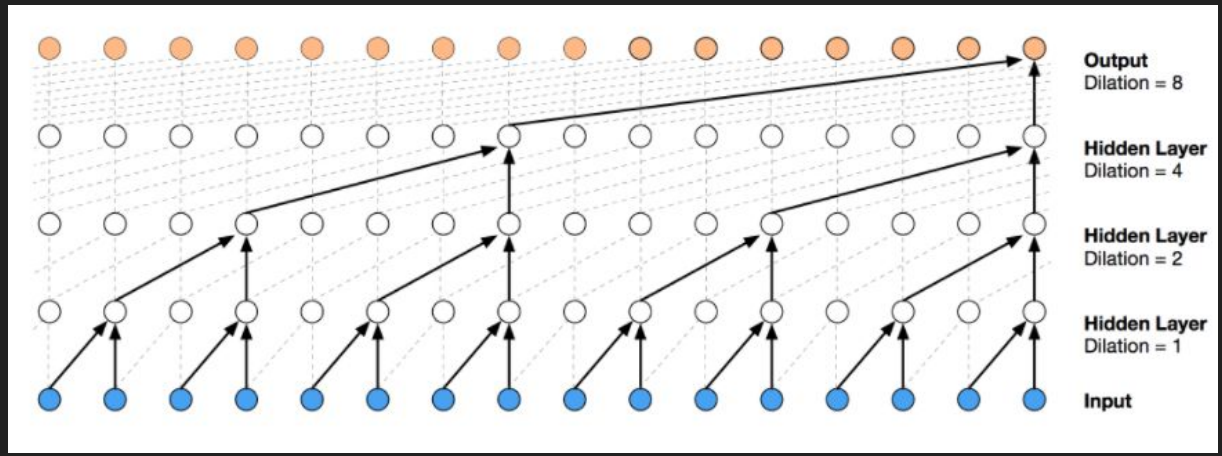
- 1D convolution through time
- Dilation of kernels in each hidden layer



Temporal Convolutional Networks(TCN)

Advantages over RNNs

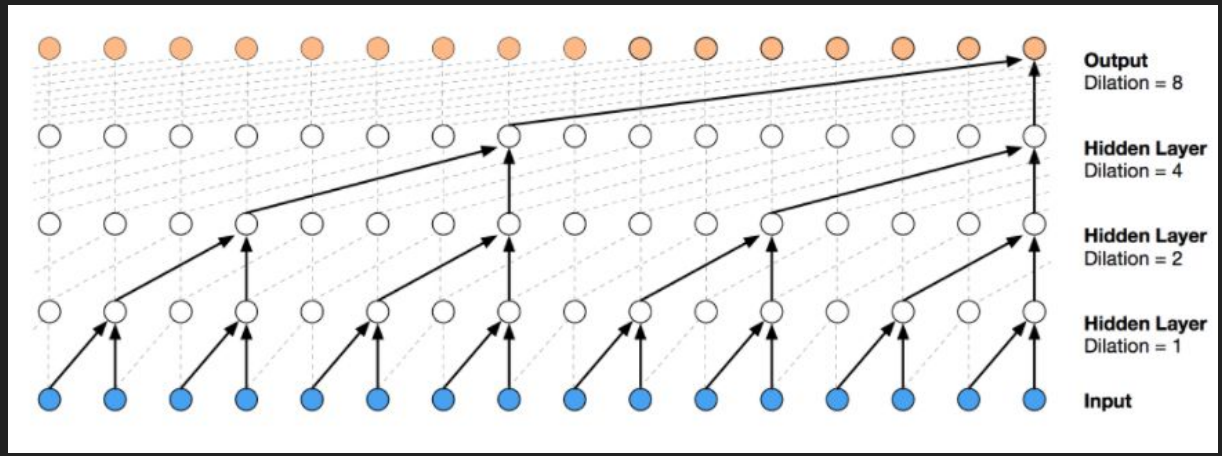
- Faster to train



Temporal Convolutional Networks(TCN)

Advantages over RNNs

- Faster to train
- No vanishing/exploding gradients

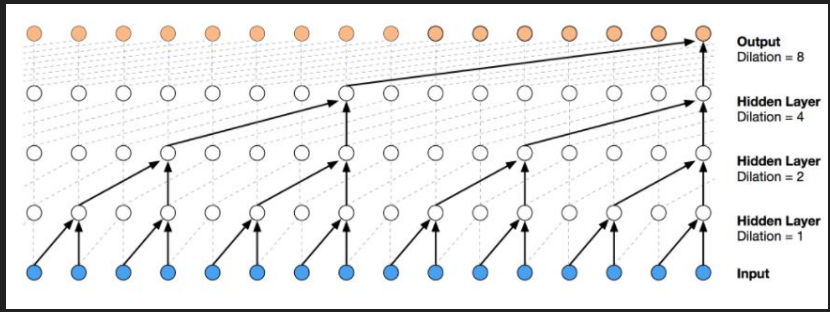


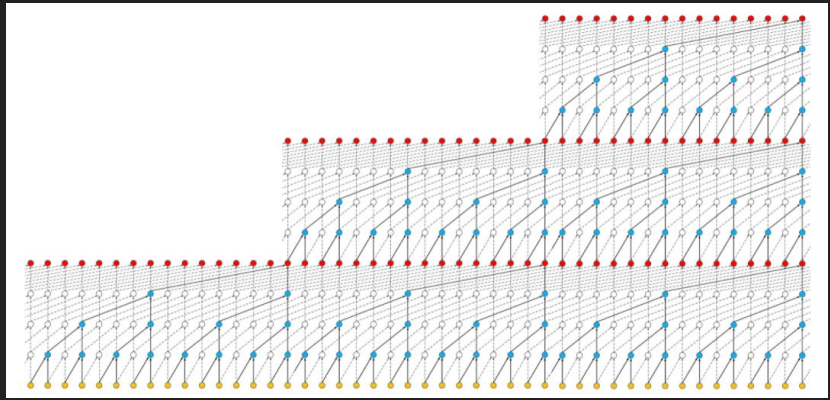
Temporal Convolutional Networks(TCN)

Advantages over RNNs

- Faster to train
- No vanishing/exploding gradients
- Tunable memory

$$R_{field} = 1 + 2\left(K_{size} - 1\right) N_{stack} \sum_{i} d_i$$

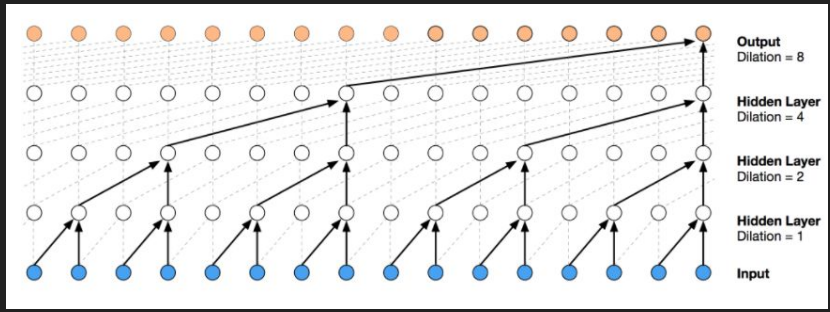


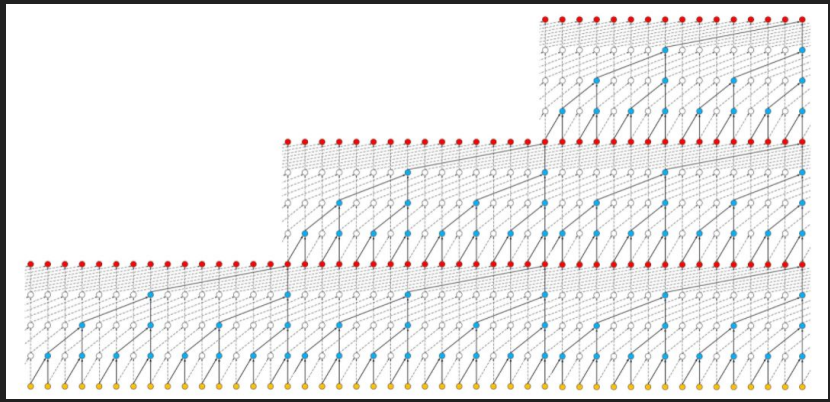


Temporal Convolutional Networks(TCN)

Advantages over RNNs

- Faster to train
- No vanishing/exploding gradients
- Tunable memory



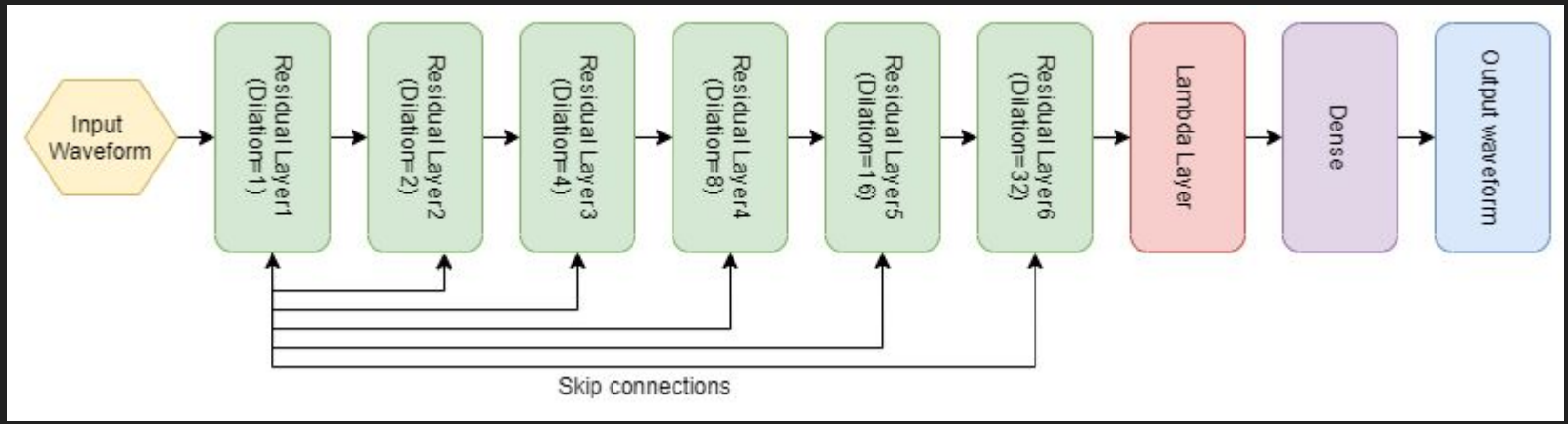


Disadvantages

- Cannot handle variable length inputs

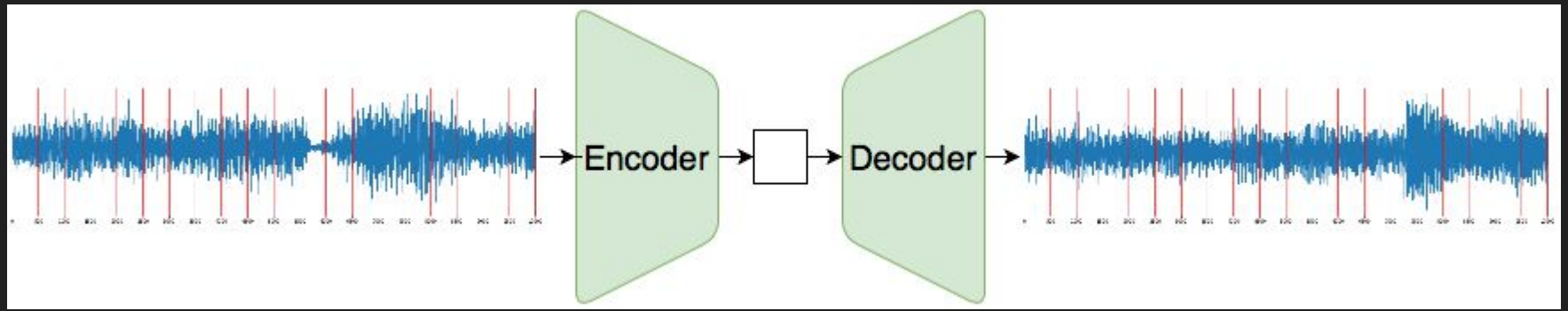
Temporal Convolutional Networks(TCN)

Architecture



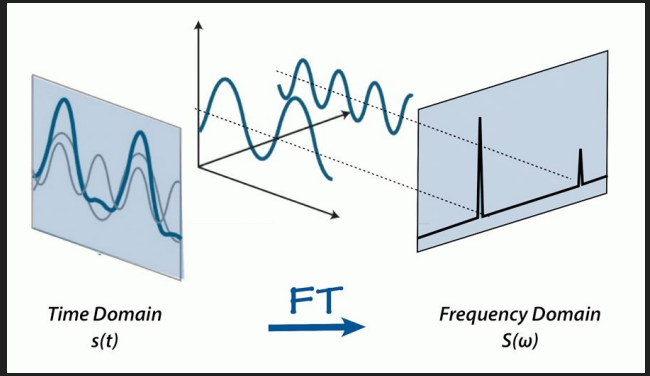
STFFT Encoder Decoder Architecture

- Deconstruct waveform into drums and melody.
- Create models for both
- Takes in 2.25 seconds of input, generates 2.25 seconds of output



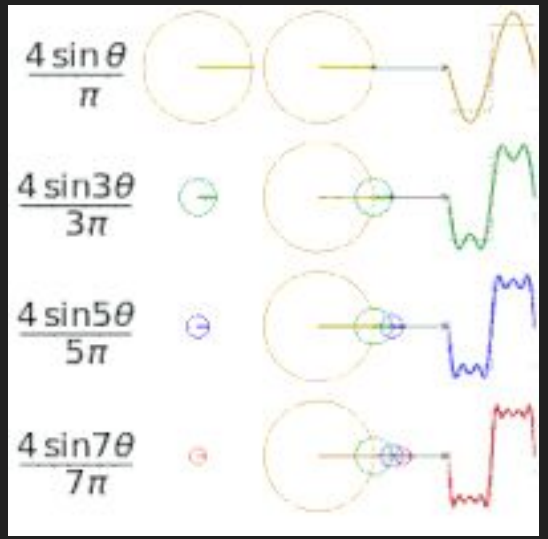
Crash Course in Fourier Transforms

- Time Domain -> Frequency Domain
- Results in more continuous and sinusoidal predictions
- Fast Fourier Transform came out in 1965, O(n log n) solution



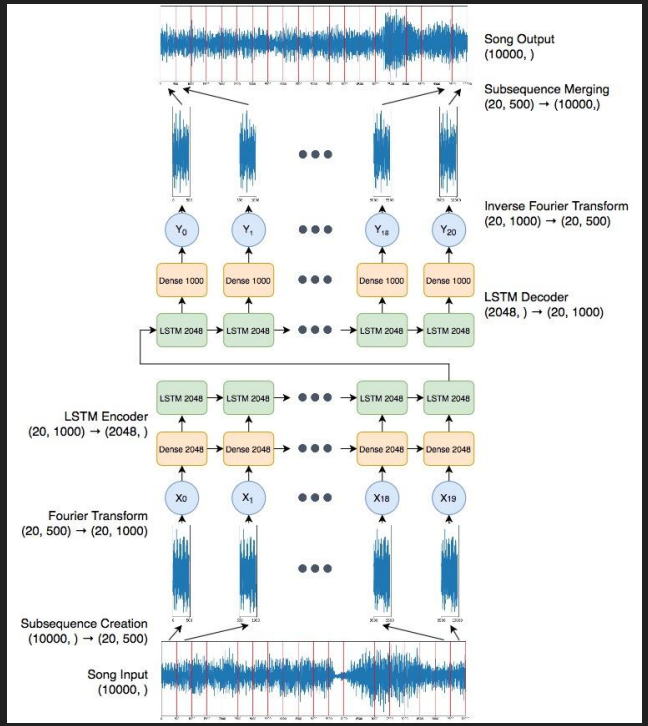
Crash Course in Fourier Transforms

- Decomposes waveform into multiple sine waves with different frequencies and amplitudes



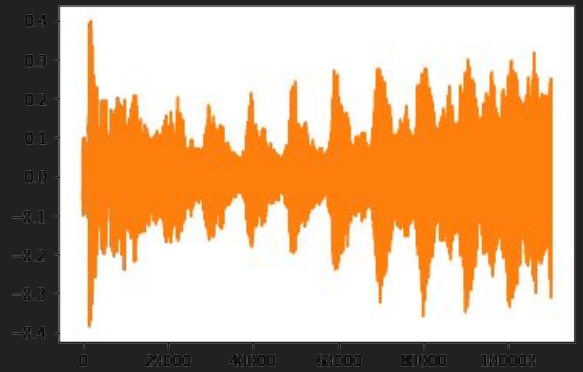
STFFT Architecture

- Utilizes FFT and Inverse-FFT
- Separates original 2 second input into 20 smaller chunks
- Encoder: Many to One
- Decoder: One to Many

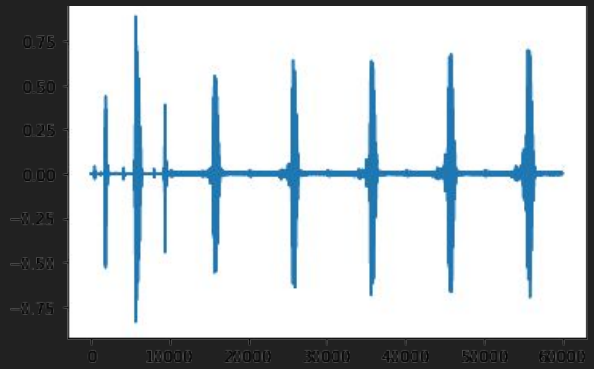


STFFT Results

Melody



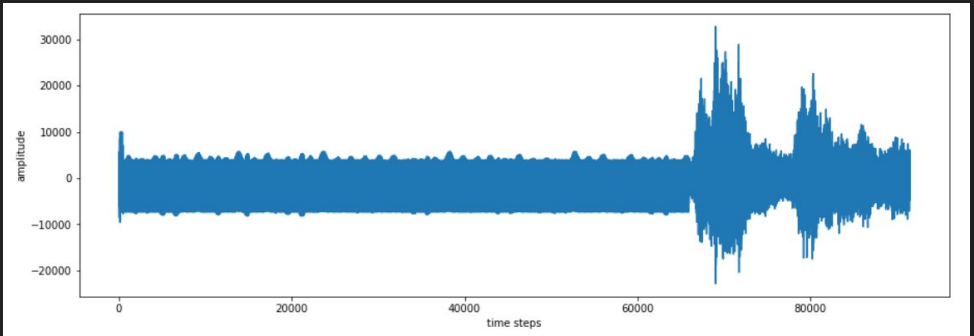
Drums



Temporal Convolutional Networks(TCN)

Results

TCN output for random noise initial input



Conclusion

What we learned

- Generating waveform data is a very difficult task.
- Need much more time and resources
- Look into different loss functions