MUSIC TRANSCRIPTION USING IMAGE SEGMENTATION

Viv Bhatia, Zach Chun, Tommy Merth, Althea Poteet | CSE 490G1 Autumn 2018

Problem

Music can be categorized as **monophonic** and **polyphonic**. Imagine someone singing 'twinkle twinkle little star'– that's monophonic. At any given point, there's only one note playing. In contrast, polyphonic music can have multiple notes playing at the same time – like chords on a piano or guitar, or choir harmonies.

Monophonic music transcription is a solved problem. But polyphonic is not! Which is unfortunate, because realistically the vast majority of music is polyphonic. Automated music transcription of polyphonic music is important for three reasons:

1. Most people are not ear trained, and find it difficult to learn a song just by listening to it. Especially with polyphonic music! Midi files and sheet music simplifies learning to play songs.
2. A transcribed version of a song makes it easier to analyze and modify.
3. MIDI files are useful for Digital Audio Workstations (DAWs), like GarageBand.

Sources

- Dataset: http://www.piano-midi.de/midi_files.html
- Commercial software alternative: https://www.lunaverus.com/cnn
- https://medium.com/@dhruvverma/music-transcription-using-a-convolutional-neural-network-b115968829f4
- UNet Paper : https://arxiv.org/pdf/1505.04597.pdf?fbclid=IwAR2FvT_Avo2S-jOSnKoLfrlnQn3DZOoEBUD8--d6Bg8qEc21bHZOMfCd4Hs
- https://github.com/EKami/carvana-challenge/blob/master/src/nn/unet.py?fbclid=IwAR25r3yunirgJbGuV-buO2vwjAp535BqYgl0OkvHZ8BrcDE1CbDhQ4bor7E

Investigation

We initially didn't know anything about the problem space. We learned:

Others begin with MIDI files. MIDIs encode notes in bars, which provide labels. The MIDIs are converted to WAVs, which are converted to images (creating spectrograms). The model then trains on the images!

A single MIDI is often processed into several spectrograms. Spectrograms are a fixed size, so creating several decreases the density of information in each.

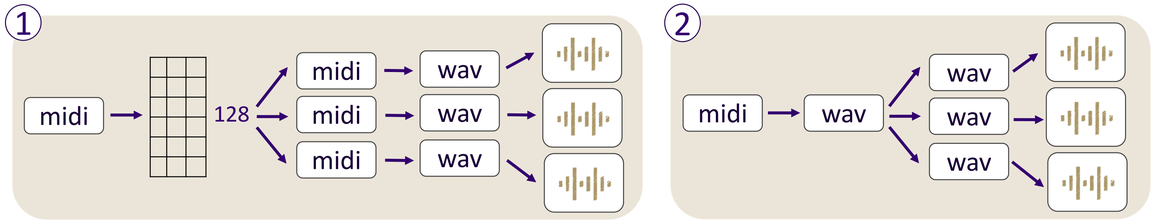
MIDI splicing is often performed by converting back and forth between a one hot vector. This simplifies creating spectrograms encoding equal amounts of time.

It's simpler to train on a single instrument. Different instruments playing the same note generate different frequency distributions. This creates different spectrograms for the same note. **We chose to train on piano MIDIs.**

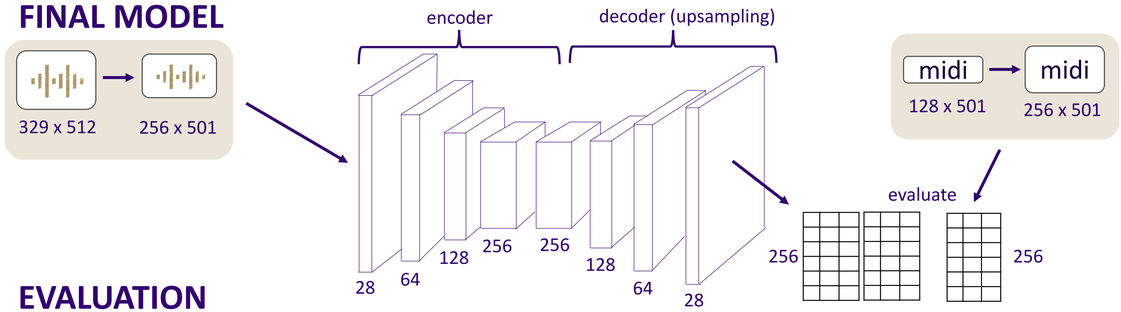
Notes are a logarithmic with regards to frequency. **Restricting the task to right hand** (higher) **notes** allows us to more easily visually identify different notes. Left hand notes are more clustered together, making them more difficult to distinguish.

Implementations

PREPROCESSING



FINAL MODEL



EVALUATION

Loss

$$-\frac{1}{N}\sum_n wt_n \log(y_n) + (1 - t_n)\log(1 - y_n)$$

F1 (accuracy)

$$\frac{2 * precision * recall}{precision + recall}$$

Precision

$$\frac{\#\ correctly\ guessed\ notes}{\#\ notes\ guessed\ total}$$

Recall

$$\frac{\#\ correctly\ guessed\ notes}{\#\ total\ correct\ notes}$$

Results & Challenges

Reshaping spectrograms and midis:

- Inputs: Preserve as much pixel information from spectrogram as possible
- Outputs: Ensure midi-one is constant multiple of # possible notes while maintaining expressive frames per second

Training on Sparse Data

- Model will predict all 0s unless we weight false negatives with high penalty.
- Use cross entropy loss with weights per class as hyperparameter