Dilated CNN + LSTM Music Style Transfer

Haojun Li[1], Jialun Zhang[2]

[1] Department of CS, Stanford University, [2] ICME, Stanford University

Abstract

Our goal is to transfer style between Jazz and Classical music in midi format. Since there are two kinds of music style transfer, we used 2 techniques to transfer style. 1) Velocity is how hard a key is hit on the piano and we can see if the same piece of music can be played differently. 2) Notes are the actual piano keys being played. We will use LSTM to train a predictor and use it to generate music.

Data and Preprocessing

Our dataset contains various sized music of both Classical and Jazz style in midi format. We have separated each music into 30 second intervals and padded the end ones. Then, we divided up the data into a chords array (how long a chord lasted), a note-on array (when a note is played) and a velocity array (with information about how hard the key was pressed). One section of **classical music** is shown below

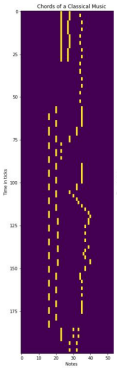

Figure 1: Chords

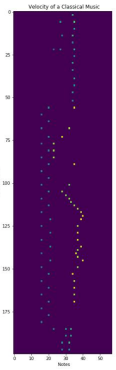

Figure 2: Velocity

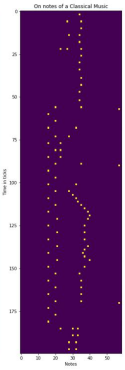

Figure 3: Note-on

Dilated CNN for Velocity

We trained 2 DCNN, one for jazz and one for classical. The diagram below demonstrates our architecture except we had much higher filter sizes rather than 2 in the diagram 4

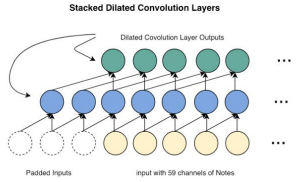

Figure 4: Dilated CNN Architecture

To help the network learn, we changed the loss function to be that we only consider it a loss if the velocity at the notes that is suppose to be played mattered (thus the network does not need to learn 0s). This is done by taking the notes-on array as a mask. The loss function for each example is shown below:

$$J(\theta) = \frac{1}{n}\sum_i 1\{\text{notes-on}_i\}(\hat{y}_i - y_i)^2$$

Then we took the chords and notes-on array of a jazz music and plugged it into the DCNN trained by classical music. Here is the result:

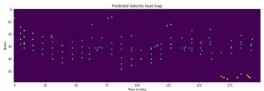

Figure 5: Predicted Velocity for Jazz

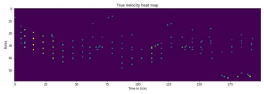

Figure 6: True Velocity for Jazz

LSTM for Chords

We want to see whether we can generate music from being 'inspired' by music of another genre. More specifically, we trained 2 LSTMs, one for jazz and one for classical. Each LSTM takes in the feature input and transforms it into its initial cell state, which condition the LSTM to behave according to the features. The architecture is below:

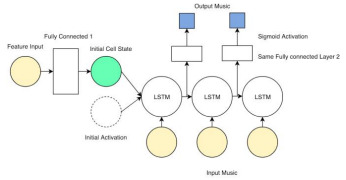

Figure 7: LSTM Architecture

We used **sigmoid** activation instead of softmax because multiple chords can be 'on' at the same time, and chose our loss function as binary cross entropy instead of categorical cross entropy for the same reason. To generate, we use the same cells but a different architecture, shown below:

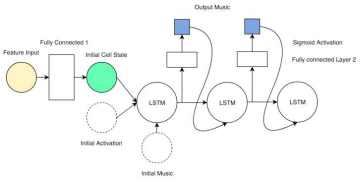

Figure 8: LSTM Generate Architecture

We run into issues with computing power. The music generated does not sound great, but we will running for longer time.

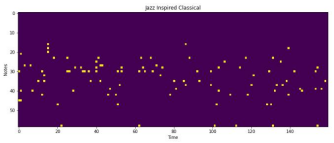

Figure 9: Jazz Inspired Classical Music

Discussion

Classical Music emphasize the higher notes more while jazz music's velocity is fairly evenly spread as you can see from figure 2 vs 6. Thus it is not surprising that jazz music played as classical music will have their high notes emphasized as you see in figure 5. We believe that the LSTM can perform better if trained longer, and current LSTM results are promising as seen in figure 9. The generated music is much more sparse due to the fact that jazz music is fairly sparsely populated comparing to classical music.

Future Work

We will definitely train more LSTM and harness more computing power to train it to get better results. The DCNN that we trained are trained on train/validation split, but in reality overfitting does not matter. So we will also try to train the DCNN on all the data and overfit it.

Reference

[1] Malik, Iman, and Carl Henrik Ek. "Neural translation of musical style." arXiv preprint arXiv:1708.03535 (2017) [2] Van Den Oord, Aäron, et al. "WaveNet: A generative model for raw audio." SSW. 2016. [3] Data from: http://imanmalik.com/assets/dataset/TPD.zip