Generate Melodies with an LSTM

Daniel Riggi

/github.com/danielriggi
/in//daniel-riggi
daniel.l.riggi@gmail.com

Background

"The music is not in the notes, but in the silence between them."
- Claude Debussy

Can deep learning models be to capture the emotion and expressiveness in music? As Debussy indicates, music is more than a simple sequence of notes. It is a combination of timing, dynamics, and sound. This presents a unique challenge, is it possible to train a recurrent neural network to predict more than the next note in a sequence, but also note duration, rests, and varying volumes.

Data

I used a collection of 74 Electronic Dance Music songs in MIDI format from a variety of artists. A few of my favorites are:

- Mako - Smoke Filled Room
- Avicii - Heart Upon My Sleeve
- Tritonal - Anchor
- Marshmello ft. Bastille - Happier

Objectives

- Featurize MIDI files in a way that is interpretable by computers, allowing the model to predict not only note sequences, but also rests, volume, and duration
- Train an LSTM to be able to generate a dance music melody
- Create music with human-like expressiveness

Data Processing

Extract Melodies/Remove Rests: The MIDI files were entire songs. I created new MIDI's with only the melodies, then removed excessive rests (greater than three seconds).
One Hot Encode Events: Every event in the song was represented as a one hot encoded array. An event was either a note on, velocity (volume), note off, or a time shift .
Augment: Each song was transposed up and down a major third, tripling the data set.

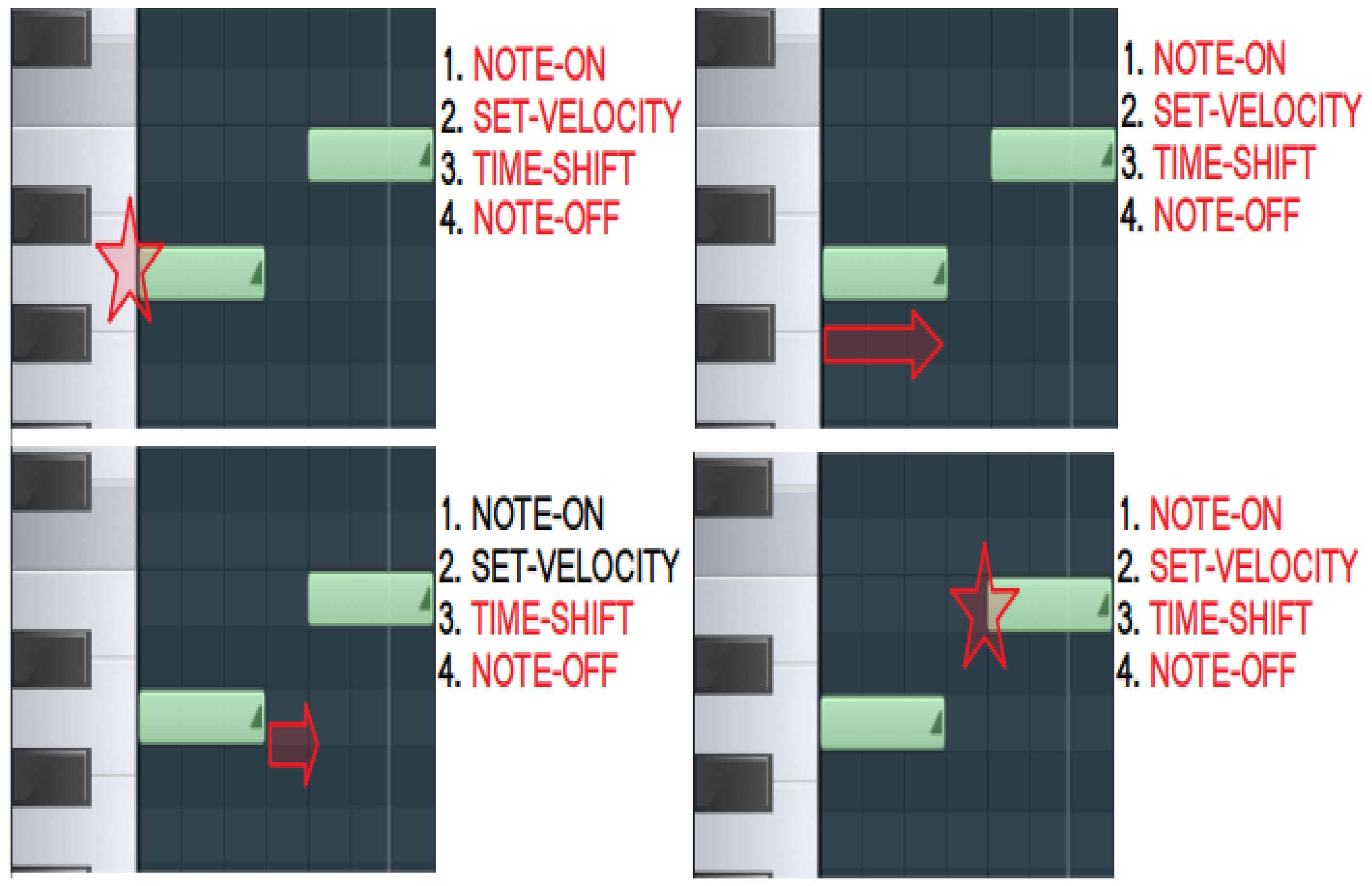

Figure 1. One Hot Encoding Representation

Model

Long Short Term Memory Network: Unlike standard feedforward neural networks, LSTM has feedback connections. It can not only process single data points but also entire sequences of data.
Architecture:
- Two Bidirectional 512 node LSTM layers
- Dropout layer with rate of 0.3
- Softmax activation function
- 200 training epochs
- Predicts next event from the previous 50

Song Generation: Songs were created using a sliding window of 50 events. The first event is generated from 50 random events. Then, the predicted event is then added to the random events, and is used to generate a new event, along with the previous 49. This continues for the length of the song, then the first 50 events are removed.

Results

Figure 2. Sheet Music after 15

Figure 3. Sheet Music after 100

Figure 4. Sheet Music after 200

Discussion

After epoch 15, the model had some strange results. It generated multiple notes that last nearly the entire length of the song. The model after epoch 100 created a wider range of notes and for more realistic duration. In addition, it played those notes with varying velocities, though this is not visible in the sheet music. Interestingly, after 200 training epochs, the model generated significantly more rests than the previous versions. This could be because it was trained on only the melodies from songs with multiple instruments. Even after removing long rests, the songs still had lots of shorter rests that were occupied by accompaniments in the full songs.

Future Work

Currently, this method is only applicable to a single instrument. Indeed, during my research, nearly every approach focused on a single instrument. I believe it is possible to expand the mechanism of one hot encoding every event in a song to include every instrument in the piece. This way, a deep learning model may be able to generate an entire song!