Emotional Music Generation

https://matthewa1999.github.io/Group11_CS4641/

Matthew Arnold
Georgia Institute of Technology

Pranav Kandarpa
Georgia Institute of Technology

Ryan Cooper
Georgia Institute of Technology

Bradford Peterson
Georgia Institute of Technology

Manas Angalakuduti
Georgia Institute of Technology

Abstract

Generating music can seem uncomplicated when there are given parameters such as rhythm, dynamics, intensity, and pitch; however, generating music from an emotion can be quite difficult. There have been many projects with similar goals, such as inputting a database of song lyrics and generating an accurate tag or genre for the song. We want to be able to generate music given a mood, based on other songs with a respective 'emotional tag' that we would have the algorithm train. First, we need to create and expand a dataset. To do this, we will be using a support vector machine to create labels of a song based on factors such as valence (musical positivity) and arousal. Next, we will train a recurrent neural network by using a database of songs' and their emotional tags. Our ultimate goal is to generate a song given an emotional tag.

1 Introduction & Background

Music generation has been attempted using neural networks since 1989, but it has gained traction with the discovery of deep neural networks' ability to learn from big data sets. As machine learning algorithms have become more widespread, and music streaming has become a larger part of people's lives, companies like Spotify and Apple have tried their hands at creating the most personalized listening experience. Spotify has recently taken a step forward in their efforts, and has patented software that could allow them to analyze user voices and recommend songs based on emotional state, accent, and other user data points. Similar music generation projects using neural networks have used long short-term memory recurrent neural networks to generate new music from previous training data. Likewise, we believe that using a Long Short-term Memory network will allow us to create pieces that resonate with people as appealing to a certain mood or emotional state. We will use a support vector machine to expand our database because they perform well with linear classification. The data is also labelled, which further supports the use of a support vector machine. The original algorithm for support vector machines was created in 1963 by Vladimir N. Vapnik and Alexey Ya. Chervonenkis.

2 Problem Definition

First the classifier must be trained. Therefore, the first input into the problem is a dataset of MIDI files and corresponding labels of valence and arousal. The MIDI files within the dataset are then properly cleaned and labeled. The newly trained classifier is used to expand our current MIDI dataset.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

The expanded MIDI dataset is then imported into the recurrent neural network. These inputs will be predefined, and each label will accompany a data set that contains a multitude of melodies with the given label. After inputting this into the neural network, the goal of this project is to output a melody that successfully replicates the mood described. We plan to use a long short-term recurrent neural network to generate a novel MIDI sequence given a data set of MIDI sequences depicting each mood. Because the perception of emotions in music may vary from person to person, certain assumptions must be made. We will make the assumption that valence and arousal will be valid indicators that can classify whether a song is happy, sad, or angry. For example, we will be assuming that a song with low valence and low arousal will be a sad song, which may not always be true.

3 Data Collection

Data collection was completed in two parts. The SVR was trained with a MIDI dataset with corresponding valence and arousal labels from https://github.com/lucasnfe/vgmidi. This github repository contains the MIDI files that our regressor was initially trained on, which was piano renditions of various video game soundtracks. Each MIDI file was parsed and a list containing the number of occurrences of notes for each file was created. This list was then standardized to ensure that every MIDI track was in C major. In addition, the repository also included two JSON files that contained a valence and arousal array for each MIDI track, which was used for the labels. Each MIDI track inside the JSON file contained a list of floating point numbers indicating valence and arousal. These numbers ranged from -1 to 1. These valence and arousal numbers were averaged for each MIDI track and are used as the response variables for the classifier. For the second part of data collection, we can currently use any MIDI file as a training sample, so we are just using the vgmidi dataset but not using the valence/arousal labels. Our idea is that we can use the regressor to generate more labeled data that we can then use for our RNN from a variety of MIDI sources with a wide range of valence/arousal content.

4 Methods

Previously, we had planned to train a regressor for the emotional labels. This would have given us a continuous scale of [-1, 1] for the valence value and a scale of [-1, 1] for the arousal value. For example, a 'happy' song with high energy could yield a valence value of 0.74 and an arousal value of 0.82. We were originally using a multi-output SVR, which looked at all of the features and predicted both the valence and arousal values, but it did not yield accurate results. Therefore, we decided to use a classifier rather than a regressor. Instead of obtaining a continuous result, we get a discrete value of 0 or 1 after rounding negative values to 0 and positive values to 1. We still uploaded each MIDI sequence file based on its respective emotional tag, implemented the LSTM recurrent neural network to train models with labeled MIDI sequence data, and test the accuracy using precision, recall, and the F1 score at which music generated can match emotion tag input. After the regressor evaluated the valence, arousal and F1 score of the midi data sets, the data sets were split into 4 based on whether they were of low or high valence and arousal. We were able to use code et al Skuldur [7] which is free to use under MIT licensing for our LSTM RNN music generation. These produced 20 songs per data set by each of the four emotions.

5 Metrics

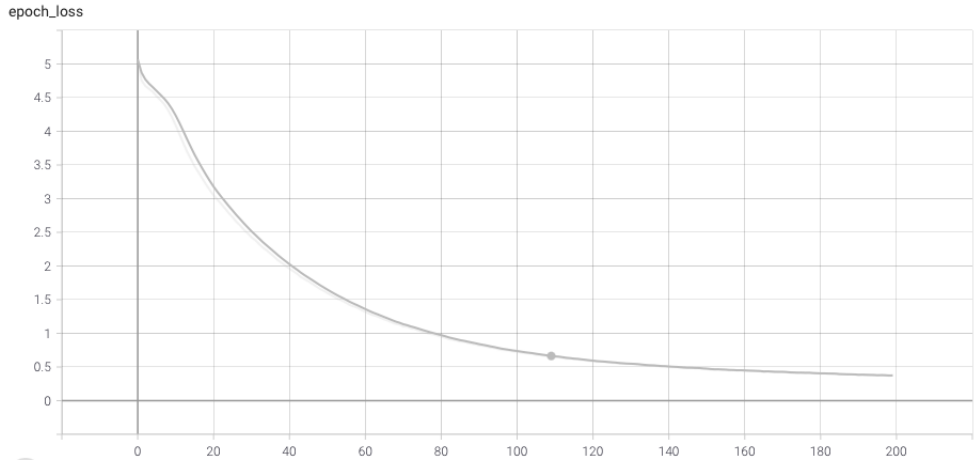
$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

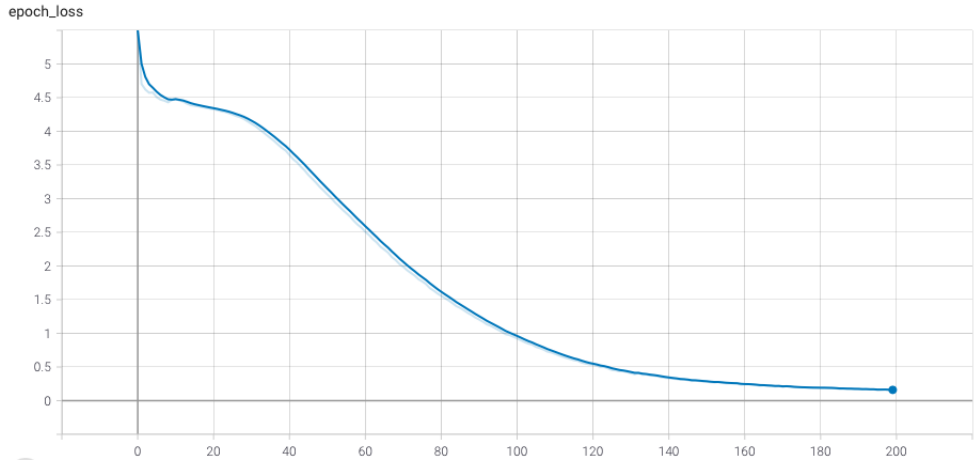
$$F1 = \frac{2 * precision * recall}{precision + recall}$$

Our SVR classifier has changed since the early stages of development. Instead of making a regression to classify valence and arousal, we instead use a binary classifier. The new metrics for this classifier are precision, recall, and F1 score.

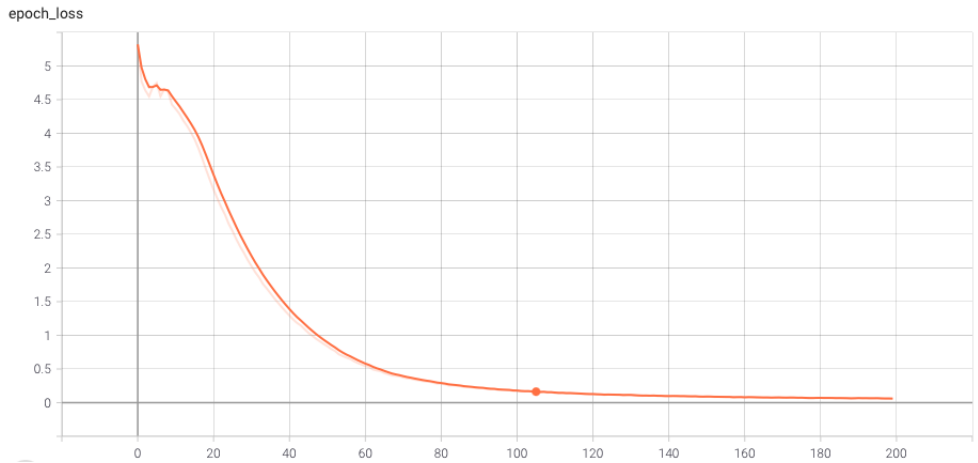
6 Results



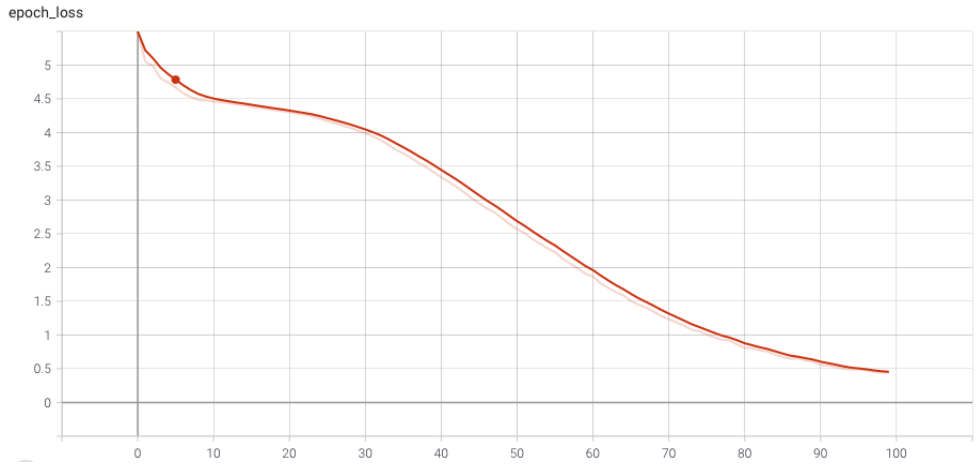
Loss plot for High Arousal High Valence



Loss plot for Low Arousal High Valence



Loss plot for High Arousal Low Valence



Loss plot for Low Arousal Low Valence

	High Energy Happy	High Energy Sad	Low Energy Happy	Low Energy Sad	Totals
Valence Acc	0.85	0.65	0.8	0.45	0.6875
Arousal Acc	0.7	0.75	0.35	0.65	0.6125
Overall Acc	0.55	0.55	0.35	0.35	0.45

For the table above, we gathered 40 files for each our emotional labels. Next, we trained four separate LSTM RNN models on each data set of music files for each emotional label. Using the model, we generated 20 total MIDI file songs for each model attributed to a different emotion. Our valence and arousal classifier was used on the generated MIDI file songs to predict the emotional category and see how well the model was able to produce music for each emotion. The accuracy was measured for valence, arousal, and overall category.

7 Discussion

We gathered 40 files for each of the four emotion labels: high valence high arousal, high valence low arousal, low valence high arousal, low valence low arousal. There were 160 files in total. Four separate LSTM RNN models were trained separately, one for each emotional folder. The goal was to produce music that would replicate the emotional label of the designated category. 20 MIDI files were generated from each of the four emotional models. After the songs were generated, we needed to use our valence and arousal classifier on the generated MIDI files to determine how well the model was able to create music for each emotion.

This brings up the necessary question: how will the performance of the model be calculated? Our team has decided to use a simple accuracy calculation. We simply compared the classifications of the ground truth music to the generated music. For example, if the ground truth's classification was [1 0 1] and the generated classification was [1 1 1] then the accuracy would be calculated as 0.67.

Similar to the accuracy statistic, low valence and low arousal music also performed the worst for the loss converge values. Low valence high arousal performed the best. This may be because low valence and low arousal music may have less defining features compared to other emotions. In the future, perhaps having more variables may help distinguish emotions better.

Overall, this experiment was a success. Our generated music was classified at a higher percentage than random chance, meaning that it is possible to generate new music solely based on an emotional label. In certain categories the classifier and generated music performed quite well. For example, high valence and high arousal music performed well throughout all of the categories. In the future,

we would hope to further refine the classifier and music generator through extensive testing and providing a surplus of training data. In addition, as stated previously, adding another feature such as tempo, beats per minute could increase the performance of the classifier and the music generator. The primary goal of this experiment was to classify and generate music based on the valence and arousal metrics, and based on our metrics, we have fulfilled this goal.

8 Ethics Statement

The intended purpose of our project was to supply music based on emotional labels attached to music files in the MIDI file format. This was accomplished by using a Long short-term memory(LSTM) neural network on the songs classified into categories with our support vector classifier. A positive social impact of this project is one in common with music. Many studies prove that music is positively correlated with a greater self-confidence, increased ability to communicate with unfamiliar people, and more participation in social activities. Our project resonates with these ideals, as this model only results in greater engagement with music due to being provided with music "matching" the emotional state they seek to be in.

A possible negative impact of our project is providing music with sad emotional labels to people who are currently struggling to improve their mental state. This can't be prevented entirely, considering the end user still retains the ability to listen to whatever songs they choose to. Our model can, however, provide them with songs with a higher valence to boost their mood. Another possible negative impact of our algorithm stems from the way we define our emotional labels. We chose valence and arousal to represent the level of energy, happiness, or sadness in songs as a sort of proxy for the actual emotional content of the songs, but these features may not represent the artists' actual intentions.

When speaking of the emotional features used in our algorithm, we should stress this point to avoid our algorithm being misused to judge or label an actual artist or their work against their wishes.

References

[1]Briot, Jean-Pierre, et al. *Deep Learning Techniques for Music Generation* Springer, 2020.

[2]L. Casini, G. Marfia and M. Roccetti, "Some Reflections on the Potential and Limitations of Deep Learning for Automated Music Generation," *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Bologna, Italy, 2018, pp. 27-31, doi: 10.1109/PIMRC.2018.8581038.

[3]Conklin, Darrell. *Music Generation from Statistical Models*, 2003, http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.3.2086.

[4]Cruz, Ricardo, et al. *I-Sounds Emotion-Based Music Generation for Virtual Environments*, 2007, citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.101.2517.

[5]Nayebi, Aran, and Matt Vitelli. *GRUV: Algorithmic Music Generation Using Recurrent Neural Networks*, 2015, cs224d.stanford.edu/reports/NayebiAran.pdf.

[6]Yang, Li-Chia, et al. *MidiNet: A Convolutional Generative Adversarial Network for Symbolic-Domain Music Generation*, Cornell University, 2017, arxiv.org/abs/1703.10847.

[7] Skuldur. *Skuldur/Classical-Piano-Composer* GitHub, github.com/Skuldur/Classical-Piano-Composer.