Generating Piano Music with Generative Adversarial Network

PAK Ka Yee, LOW Zhi Hao

Background

Each composer has their personalised style of making music, and these styles are not kinds of knowledge that can be easily transferred between people. This music's inherent pattern needs to be studied thoroughly, and not a single methodology can be applied to all kinds of music styles or artists.

Objective

1. Compose realistic and enjoyable piano music.

2. Real-time adjustment of the music tunes, rhythm and loudness.

Motivation

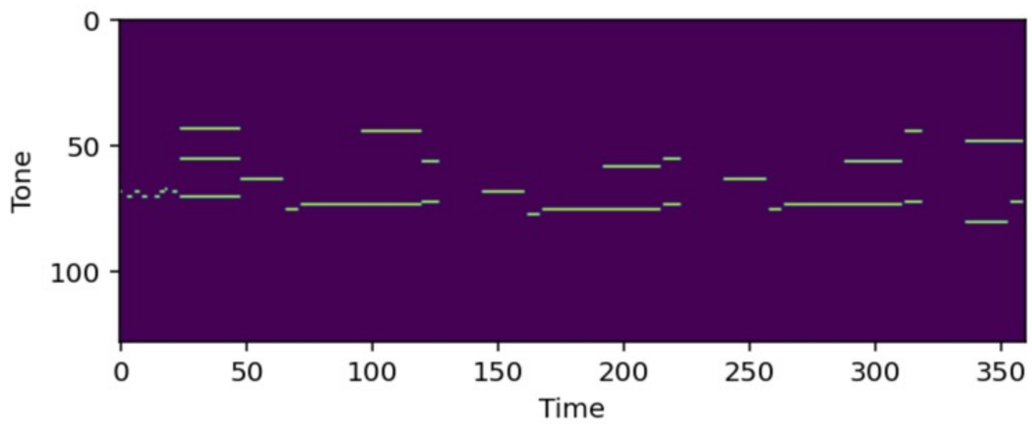
Real-time music generation is costly to work to be done by hand. Sound generations will definitely open a new dimension for incredible experiences.

In this project, we hope to simulate piano music with instrumental-only kinds of music.

Dataset

We obtain the dataset from Kaggle with namely Classical Music MIDI by Soumik Rakshit

MIDI is a format that describes music in tracks, each track has time and tone dimension. As an example, below is the illustration of the MIDI format.



But each MIDI have different tempo at different time, this will make the complexity of the application high, therefore we decided to use 120 beats/second. And we segment the data into 120 beats x 128 tones.

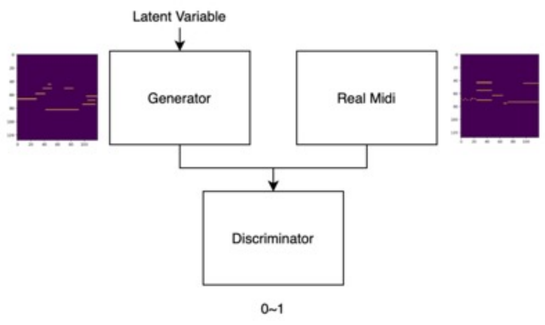
Model Design

There are two agents in GAN, the first is the generator. The generator takes in the random vector of length 100 and outputs the MIDI 2D matrix by using transposed convolutional neural network.

The output of multiple channels are then combined into one channel, same as the dataset.

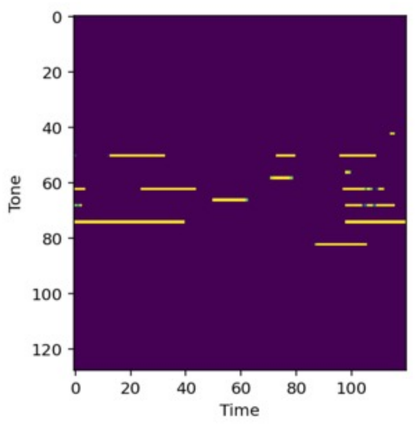
The second is the discriminator. The discriminator goal is to identify if the input is real or generated from the generator. The discriminator is therefore a simple classifier. The loss employed in both model is binary cross-entropy and the optimizer is adam.

So for training, we will train the discriminator as normally, and for the generator we will freezes the discriminator and train on the GAN network to update the weights, below shows the GAN Network.

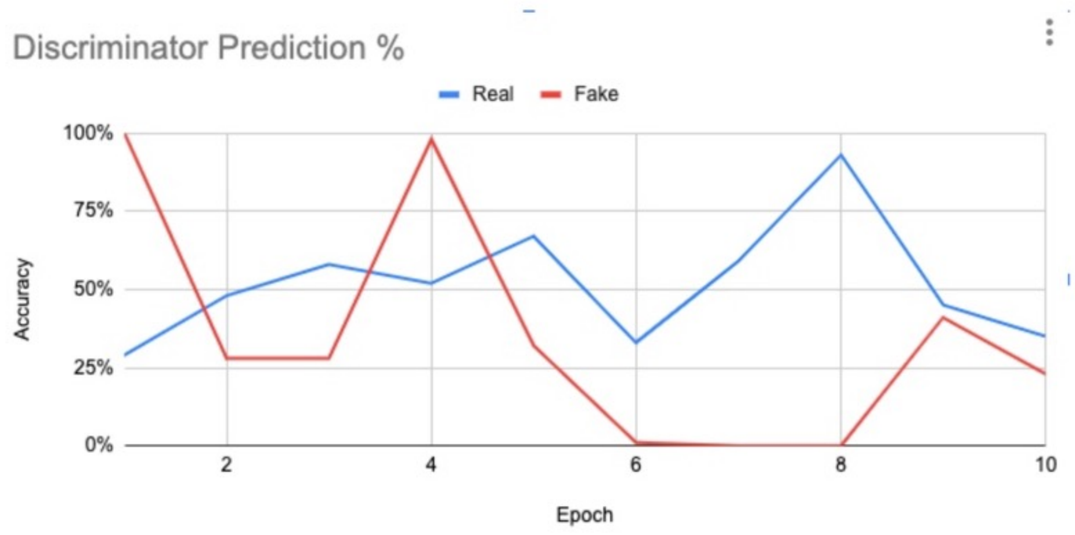


Result

As the result, we generated a piece of music for 1 second as an example



We then show that the convergence of accuracy on discriminator.



As we can see at Epoch 9 and 10 the Discriminator starts to converge to 50% accuracy.

Conclusion

A Generative Adversarial Network can be used to generate music, and Real-time music generation can be done by tuning the latent variable. The music melody will follow the tuning of the latent variable to slowly change following the desired result.

Rakshit, S. (2019, May 17). Classical music MIDI. Retrieved April 23, 2021, from https://www.kaggle.com/soumikrakshit/classical-music-midi