Name: Jason Zhou
Mentor: Dr. Dongjin Song
Status Report #: 21
Time Spent on Research This Week: 2.5
Cumulative Time Spent on Research: 162.25
Miles Traveled to/from Mentor This Week: 0
Cumulative Miles Traveled to/from Mentor: 0
===

Monday, March 7th, 2022: (0.5 Hrs)

Last week, I ran my neural network program and calculated the loss values. For some reason, they were incredibly high, so I asked Dr. Song about my problem. He said that I had probably forgotten to normalize my data, and he was right.

During the process of converting sound files into Mel spectrograms, I completely forgot to normalize the values. He let me know that the equation to normalize the data was (x - m)/s. Where x is the data, m is the mean, and s is the standard deviation.

Tuesday, March 8th, 2022: (0.5 Hrs)

After my meeting with Dr. Song, I worked on normalizing the data, which meant that I had to calculate the mean of the time series data (which is audio converted into a numerical form where each number represents the amplitude at a certain time). As such, I loaded up all of the time-series data into one array and took the mean and standard deviation by using functions from the numpy library.

Although writing the code did not take me that long, it took an incredibly long time to load all of the values. For reference, each second of audio has 22,050 samples[1]. Each file is 10 seconds long, meaning that there are 220500 samples (or pieces of data) in one file, and there are a total of 47,152 files. This means that my computer had to process over one billion pieces of data.

My computer incredibly slow (because all of the memory was taken up), so I tried to restart it. However, for some reason, my computer decided to do a windows update during the restart. Doing a windows update right after processing over a billion numbers was simply too much for my computer, and it crashed.

Wednesday, March 9th, 2022: (0.5 Hrs)

---

[1] A sample is a number that represents the amplitude at that time. In this case, each second had 22,050 samples; however, sound is technically continuous, so, theoretically, one could take an infinite number of samples from a single second of audio. 22,050 samples just happens to be the default that most people use.

At this point, my computer seems to be broken. I am pretty sure that while trying to do a windows update, the crash caused some system files to become corrupted, which has greatly hindered the computer's performance. I found this out while trying to process more data. Specifically, the current Mel spectrograms that I have right now are 10 by 474 (length the array); however, I wanted to change it to 10 by 431 to see if that would help the neural network train better.

Although the data was able to process, it took a very long time and slowed the other processes that were running on my computer (chrome, Spotify, etc). This normally would not have happened.

currently have too much going on, so I will fix my computer later.

Thursday, March 10th, 2022: (0.5 Hrs)

I ran the version of my training data through my neural network. I did not use the window sliding technique because that would require me to multiply the amount of data I was using to train my network. I was afraid that such a large amount of data would crash my computer again, so I simply went with inputting each Mel spectrogram as a whole.

Overall, it ended up doing well. Specifically, the loss values came down to 35.0865, which is the lowest its ever been.

```
Epoch 0, Loss: 165.3579
Epoch 1, Loss: 60.1968
Epoch 2, Loss: 44.4041
Epoch 3, Loss: 42.5446
Epoch 4, Loss: 41.6991
Epoch 5, Loss: 41.1762
Epoch 6, Loss: 40.7497
Epoch 7, Loss: 40.4290
Epoch 8, Loss: 40.1832
Epoch 9, Loss: 39.9938
Epoch 10, Loss: 39.8363
Epoch 11, Loss: 39.6938
Epoch 12, Loss: 39.5616
Epoch 13, Loss: 39.4455
```

(This is a picture that shows the starting loss for my model. Basically, it shows how well my model was at the very beginning of its training. The higher the value, the worse it did. For reference, anything above 100 is astronomically bad)

```
Epoch 237, Loss: 35.0986
Epoch 238, Loss: 35.0976
Epoch 239, Loss: 35.0964
Epoch 240, Loss: 35.0956
Epoch 241, Loss: 35.0947
Epoch 242, Loss: 35.0936
Epoch 243, Loss: 35.0927
Epoch 244, Loss: 35.0916
Epoch 245, Loss: 35.0906
Epoch 246, Loss: 35.0896
Epoch 247, Loss: 35.0885
Epoch 248, Loss: 35.0874
Epoch 249, Loss: 35.0865
```

(This is a picture that shows the ending loss for my model after 250 epochs or rounds of training. Compared to the starting loss of 165.3579, the model has greatly improved and now has a loss value of 35.0865.)

Friday, March 11th, 2022: (0.5 Hrs)

As I had recently been changing how I was preprocessing my data, I wanted to see how the number of melbins[2] would affect my neural network.

Although I was able to process the data, I again had trouble with my computer slowing down. In the end, my computer crashed again.

Sunday, March 13th, 2022:

I have not counted this day as part of my research project. However, I did just want to say that I was able to fix my computer. After a refresh install of Windows 10, my computer is now up and running again.

References:

N/A

[2] The more mel bins there are, the more height is added to the Mel spectrogram. Additionally, the more melbins there are, the more accurate the conversion from sound is. The rationale for this is that you are using more values from a continuous piece of data(sound). The more discrete points you plot to model a continuous piece of data, the more continuous it will look.