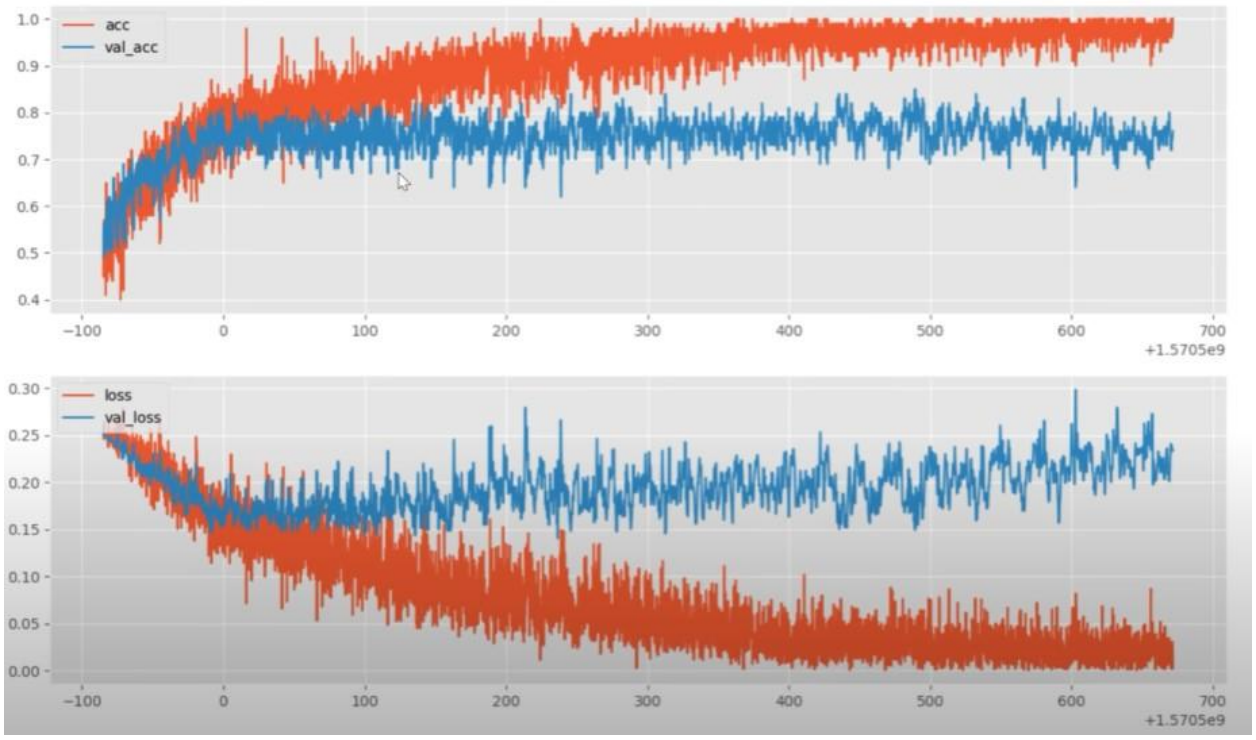
Name: Jason Zhou
Mentor: Dr. Dongjin Song
Status Report #: 17
Time Spent on Research This Week: 4.5
Cumulative Time Spent on Research: 125.5
Miles Traveled to/from Mentor This Week: 0
Cumulative Miles Traveled to/from Mentor: 0
==========

Monday, January 31st, 2022:

I was unable to meet with my mentor today due to some conflicts he had. Unfortunately, because of my schedule, we were unable to schedule a makeup meeting for this week. Thus, we will simply have to resume our weekly meetings next week.

Tuesday, February 2nd, 2022: (1 Hour)

On this day, I finished the video series I had been watching on how to create neural networks in PyTorch. The last episode went over model analysis and how to tell if one's model still needs to be trained. He stated that there are essentially four factors to determine this: in-sample accuracy, in-sample loss, out-of-sample accuracy, and out-of-sample loss[1]. These four attributes can be graphed to see how much the network has progressed in its training.



[1] In-sample data is the data that is used to train the neural network. Out-of-sample data is data that the model was not trained on but is being asked to evaluate (essentially, real-world data).

(This is a picture of the graphs I mentioned above. One graph measures training and testing accuracy, and one graph measures training and testing loss. I am not quite sure what the x and y axes are; however, one only needs to pay attention to when each diverges. The divergence indicates if a model needs to be trained further or not)

Wednesday, February 3rd, 2022: (1.5 Hour)

The video series gave me the knowledge I was previously missing to preprocess my data. So, now that I finished the series, I began preprocessing my data.

I first constructed a for loop to iterate through the array of addresses I had created. Then, I converted each audio file into a Mel spectrogram and stored them into a Python list. I first started by converting a small sample of files to make sure that the process actually worked. Afterward, I moved on to processing all of my data, which consisted of over 40,000 files.

```
In [*]: #loading all of the melspectrograms into the array for the training data
        #which will later be inputted into the neural network
        for file in tqdm(range(len(addresses))):
            scale, sr = librosa.load(addresses[file]) #taking the sample rate and time-series data of the audio
            #converting the aboe data into a melspectrogram, which graphs frequency over time
            melspectrogram = librosa.feature.melspectrogram(scale, sr=sr, n_fft=2048, hop_length=512, n_mels=10)
            logMelspectrogram = librosa.power_to_db(melspectrogram) #converting the melspectrogram into a log melspectrogram
            training_data.append(logMelspectrogram) #adding it to the training dataset
            #in this case, the data does not need to be split into testing and training data because all audio was taken from the
            #training folder of the DCASE Data

          0%|          | 87/47151 [00:30<4:54:18,  2.67it/s]
```

(Here is a snapshot of my code that I used to iterate through my data. Overall, the process took over seven hours.)

Thursday, February 4th, 2022: (2 Hours)

Everything was going well. All I had to do was save this data to a text file so that I would not lose it. However, at this point, I ran into a problem. Because all of the audio files were different durations, that meant that each Mel spectrogram was slight different in size. Apparently, NumPy, the module I was using to convert my tuples [2]into text files, is unable to handle a collection of tuples that are of different sizes.

It was at this point that I remembered that I probably had to pad the data. This is simply when you artificially increase the size of an image (in this case the Mel spectrogram) so that it conforms with the sizes of the rest of the data. At the moment, I am unsure of how to do this, so I will have to look into that next week.

[2] Tuple is simply a method of data collection

References

sentdex. (2019, October 15). *Model analysis - deep learning and neural networks with Python and Pytorch p.8* [Video]. YouTube. https://www.youtube.com/watch?v=UuteCccDXCE&list=PLQVvvaa0QuDdeMyHEYc0gxFpYwHY2Qfdh&index=8