Amy Turner

Github Link: https://github.com/amymt56/CS4347

Architecture:

Used PyCharm with Anaconda Environment. Used Keras, Tensorflow, and PyTorch packages, as well as everything else in requirements.txt

 - LEARNING RATE

I didn't see much changes by changing the learning rate as I thought I would. I saw that accuracy increased slightly when reducing the learning rate. I decided to reduce it from .0001 to .00001, which caused a very slight increase in average accuracy

 - SCHEDULER (IF USED)

Not used

 - OPTIMIZER

I used RMS

 - EPOCH

I found that decreasing epoch caused accuracy to start high, but decrease between 3 out of the 5 tests. However, the average accuracy still ended up being 83.414%. When increasing the epoch to 15, accuracy went up to 86.8%. However, increasing epoch could lead to overfitting, so it is safer to find a middle ground, which is why I decided to keep my epoch at 10.

 - TRAIN SIZE

I saw an increase in accuracy when training size increased, so I doubled the training size. I was worried about overfitting, so I did not increase it beyond that.

 - TEST SIZE

I found that there wasn't much of a difference in accuracy when there were different test sizes, so I kept it at the same as the train size to ensure a 50 50 split

Final Results

Final accuracy: 92.17%

Final Loss: 19.82%

Final Results

Accuracy	Loss	Val Loss	Val Accuracy