**Question #2**

Summary

During the entire session we've attempted three different approaches

-Decreasing the batch size and adding one more layer

-Adam optimizer

-RMSprop optimizer

Throughout the entire session we've gained results around 85-95 accuracy, overall the loss was in the same level and finally decided that the best among them is the Adam optimizer.

All contains the same amount of data, 25,000 for train and 5000 for test and shuffled data.

Test #1

Amount of layers : 5

Optimzer : RMSprop

Loss function: Categorical Cross Entropy

Batch : 128

Epochs : 100

Dropout : 0.5,0.25,0.5,0.5,0.5

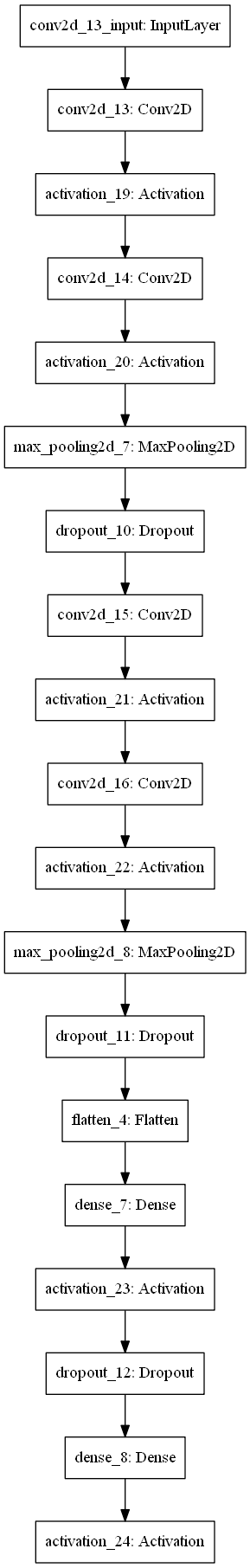
Activitation : Relu

Results :

Accuracy : 89.2%

Loss : 0.2

Next page contains the layers architecture.



Test #2

Amount of layers : 5

Optimzer : ADAM

Loss function : Categorical Cross Entropy

Batch : 128

Epochs : 100

Dropout : 0.5,0.25,0.5,0.5,0.5

Activitation : Relu

Results :

Accuracy : 97.2%

Loss : 0.32

In thie paticuler test we've tried to use the Adam optimizer. We've gained a better result than beforehand, therefore the ADAM optimizer performed better this time.

(The above written can vary, depends on how the data had been shuffled, etc…)

Test #3

Amount of layers : 6

Optimzer : Adam

Loss function : Categorical Cross Entropy

Batch : 64

Epochs : 50

Dropout : 0.5,0.25,0.2,0.2,0.5

Activitation : Relu

Results :

Accuracy : 88.2%

Loss : 0.47

In this test we've attempted to add another layer with the following hyper-parameters :

Activitation = relu

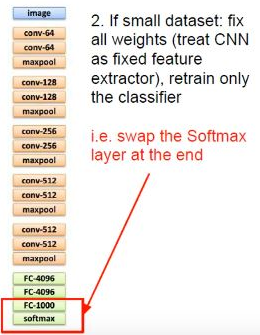
MaxPooling2D(pool\_size=(2, 2))

Dropout(0.2)

The accuracy has dropped and the loss raised, therefore we stick with test number two.

Transfer learning :

During the research we've learned that the methods to handle transfer learning can vary depends on the size of the dataset and how different the current the dataset from the desired one we'd like to train on. In our case the dataset is small and the dataset we'd like to train on is very different from the current one, we froze all the layers besides the last one and then removed the last one, added another new layer with the trainable parameter to true and trained the new net with the new data and the old data combined.



Amount of layers : 5

Optimzer : ADAM

Loss function : Categorical Cross Entropy

Batch : 128

Epochs : 100

Dropout : 0.5,0.25,0.5,0.5,0.5

Activitation : Relu

Results :

Accuracy : 80%

Loss : 0.4