# Chest X-Ray Classification with ML

# **Introduction**

According to the NIH1, Pneumonia is a general term for infection of the lungs which can range in severity from mild to life-threatening. Diagnosis of pneumonia is often done using chest x-ray imagery. To the trained eye, there are visual cues to distinguish between healthy lungs and lungs affected by pneumonia. This can be treated as a binary image classification task, which motivates training an effective machine learning model. This report will detail my attempt at training such a model.

# **Dataset and Related Work**

The dataset used for this was obtained from Kaggle2, which cites data obtained from MendeleyData3. For the model, I referenced other models by accessing the “code” tab of the dataset on Kaggle including but not limited to “chest-x-ray-xception-944” and “Pneumonia Detection using CNN(92.6% Accuracy)5". Most of the code used for data and image processing was directly obtained from “chest-x-ray-xception-94,” with adjustments for clarity and for data/source management.

# **Methodology**

There were several key choices made in building this model. Images were pre-processed by normalizing their resolution to 256x256x3 in order to ensure uniformity. The training set of 5216 was split into a training set and a validation set of size 4695 and 521 respectively. The given validation images were discarded due to the lack of data (there were a total of 16 images in the “val” folder).

Four hidden layers were chosen. The choice of depth of layers was somewhat arbitrary since ~~I have no idea what I’m doing~~ I lack the experience to intuit the complexity of the problem that is being tackled. The first three layers were chosen to be convolutional for educational purposes regarding implementation and because they are more efficient for image classification due to the nature of the data format. The number of layers chosen was also somewhat arbitrary; sources online recommended some binary exponent. The number of filters was doubled at each layer to reflect the complexity of patterns represented at subsequent layer depths.

Each layer first utilized batch normalization for stability activation. ReLU was used as the activation function for all four layers. For the three convolutional layers, a max pooling size of 2x2 was used to reduce spatial dimensions, and finally dropout was applied at a rate of .25 as a measure to prevent overfitting.

The final hidden layer was a fully-connected layer with 300 neurons. This was sort of an arbitrary choice; it seemed to be common in many of the models I browsed through while doing this project, so I wanted to see how it would perform. As with the convolutional layers, batch normalization was applied, and ReLU was used as the activation function. Dropout for the hidden layer used a rate of .5. This was yet another purely experimental choice that fell within conventional parameters and seemed to reflect other models.

# **Experimental Setup**

The training input of our model consists of images of varying sizes produced via Labeled Optical Coherence Tomography (OCT) and Chest X-Rays. Images are pre-processed by normalizing pixel values to a resolution of 256x256 pixels with 3 color channels, making the overall input 256x256x3. The training set included 5216 files, which was split into a training set and a validation set at a 9:1 ratio.

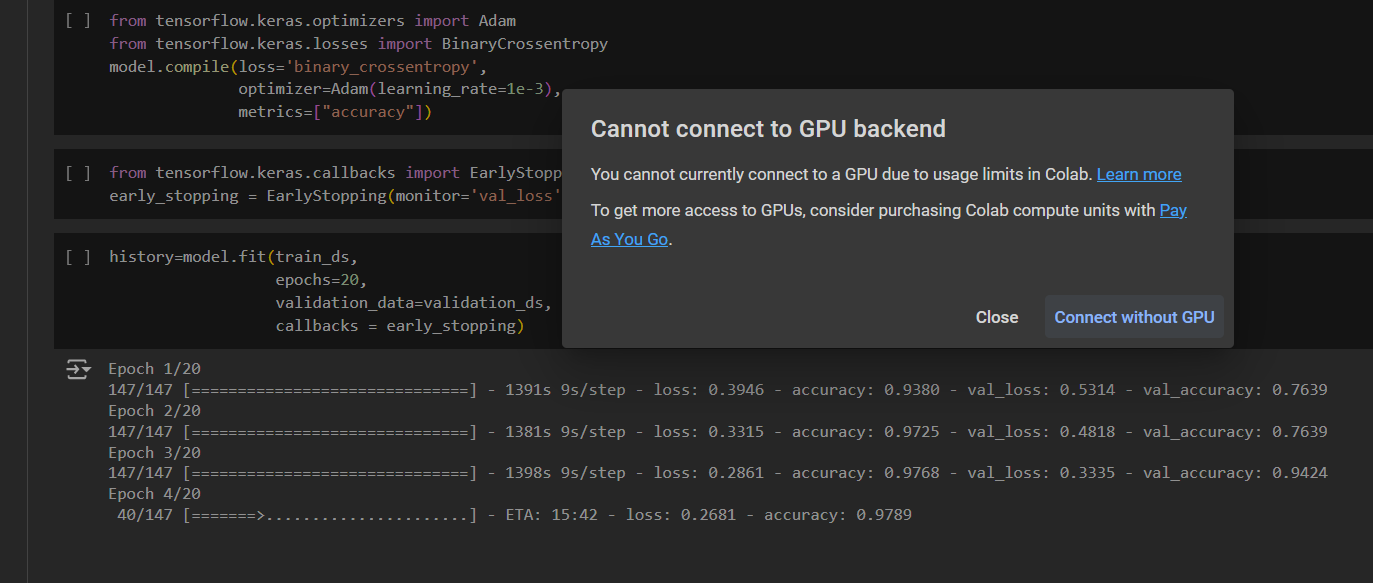
The model has four hidden layers, of which the first three are convolutional layers. The first layer has 32 filters, the second layer has 64 filters, and the third layer has 128 filters, all with the same kernel size of 3x3. Batch normalization is applied at the beginning of each layer. Each layer is then activated by ReLU to introduce non-linearity. Max-pooling with a pool size of 2x2 size is applied following activation, and finally dropout with a rate of .25 was applied. After the three convolutional layers, there is a fully connected layer with 300 neurons. Batch normalization is applied followed by a final ReLU activation before the output layer is normalized and activated using a sigmoid function.

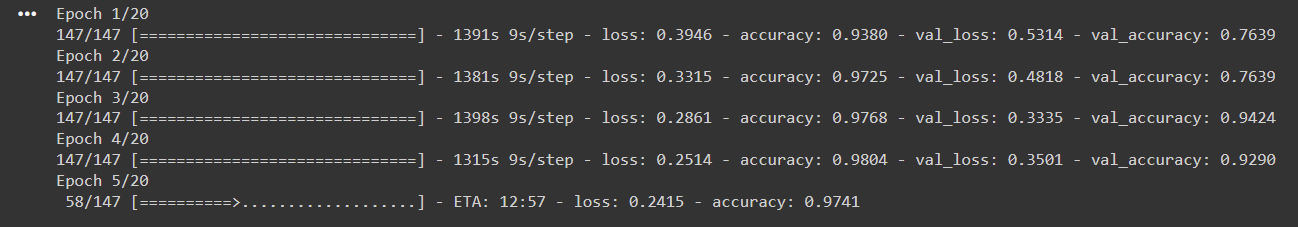
The model was compiled using Binary Crossentropy loss since our output demands binary classification. The optimizer chosen was Adam with a learning rate of 1e-3 because this is the one we used in labs and I don’t know any other optimizers. Additionally, early stopping callbacks were used for fitting the model with a patience of 5 epochs for the validation loss in the event to prevent overfitting and hopefully obtain a more generalizable model.

# **Measurement**

Some measurements worth noting include the distribution of training data - around 25% of the 5216 files of training data were labeled as “normal” i.e. healthy lungs, with 75% being pneumonic. The distribution of categories of the 624 files of test data was 37.5% “normal” to 62.5% “pneumonia.” I did not apply any corrections to the weight of parameters, but it is worth noting that the efficacy of the model could have been impacted due to this bias.

# **Results Analysis, Intuitions and Comparison**

Unfortunately due some time limitations and some unfortunate circumstances (image included), this model was not fully trained, so this discussion will be limited to available data and corrected in an errata section in the near future.

At the point of writing this report (about 30 minutes before the deadline). This the current state of the model being trained:

As we can see, there is a trend of significant decreases in loss for both the training and the validation datasets, with the corresponding increase in accuracy. Since this is early on in the training of the model, it is inappropriate to draw conclusions from the current results. From comparison with the results of other models described on Kaggle, it appears that these are relatively good measurements at this point in time.

# **Conclusion**

Though there are limitations, machine learning models have great potential for classification tasks like x-rays as a tool for diagnosis. Machine learning models could be an effective supplement to medical treatments by saving time and resources, especially for common conditions such as pneumonia with relatively easily accessible diagnostic methods such as x-rays.

# **Contribution**

This project and report was done individually by Victor Lee. Most of the data-processing code was obtained from another model used to train this data as described above4.

**References:**

1. <https://www.nhlbi.nih.gov/health/pneumonia>
2. <https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia/data>
3. <https://data.mendeley.com/datasets/rscbjbr9sj/2>
4. <https://www.kaggle.com/code/abdmental01/chest-x-ray-xception-94>
5. <https://www.kaggle.com/code/madz2000/pneumonia-detection-using-cnn-92-6-accuracy>