Chest X-RAY Image Classification for Identifying COVID-19

Jack Carson

CPSC 4300

Applied Data Science

Spring 2023

Introduction

In this project, my main question was whether it is possible to accurately predict the presence of COVID-19 in chest x-ray images using machine learning models. The motivation for this project stems from my personal experience with my brother, who was not correctly diagnosed with COVID-19 from his chest x-ray images in the hospital. This experience made me realize the importance of accurate and timely diagnosis of COVID-19, which can potentially save lives and reduce the burden on healthcare systems.

For this project, I utilized data from a Kaggle repository (https://www.kaggle.com/datasets/prashant268/chest-xray-covid19-pneumonia) containing pre-labeled train, test, and validation sets of chest x-ray images for normal, COVID-19, and pneumonia. The data repository contained approximately 1.5GB of .jpg images, all of which were pre-labeled as being either positive or negative using PyTorch labeling. Each .jpg image was roughly 400-500 kB and was converted into a 256x256 image tensor for ingestion into the convolutional neural networks. Only normal and COVID images were used during this project.

The aim of this project was to develop a model that accurately predicts the presence of COVID-19 in chest x-ray images, which can potentially provide a faster and more cost-effective solution for diagnosis. Additionally, this project aimed to explore the importance of accurate image data, the pipeline by which images can be converted into tensors for use in convolutional neural networks, and the types of networks that can perform well for image classification tasks where COVID-19 positive images may have very fine pixel densities that may distinguish them from normal images.

Summary of Exploratory Data Analysis

The unit of analysis in this dataset are the labeled images, which are digested as 256x256 image tensors with 3 color channels. There are 1650 total images in the local dataset, with 585 positive and 865 negative images used for training, 75 positive and 75 negative images used for validation, and 20 positive and 30 negative images used for testing after model training.
The dataset contains X-Ray images from early 2021 to mid 2022 and has been downloaded for Kaggle projects over 10,000 times since its release in May 2022.

To clean the data, the images were separated into separate labeled directories for training, testing, and validation. Only “PA” X-Ray images were used, which presents the chest in anatomical position, and images were resized to be 256x256 and centered on the center of the chest (Area of Interest).

For visualization of the response, templates created by the model for positive and negative images were generated. To visualize key predictors against the response, scatterplots were used to examine the relationship between image intensity and the presence of Covid-19.

The results suggested that there were slight differences in pixel intensities between positive and negative images, which may be important predictors in explaining the presence of Covid-19. Overall, the EDA helped to identify key predictors that may be important for predicting the presence of Covid-19 in chest x-ray images, which can inform the development of machine learning models for accurate diagnosis.

Also included in the supplemental repository (https://github.com/ieee8023/covid-chestxray-dataset) was a metadata spreadsheet containing some categorical features of each chest X-Ray. Below are some images generated during the EDA using this metadata, along with some example X-Ray images.

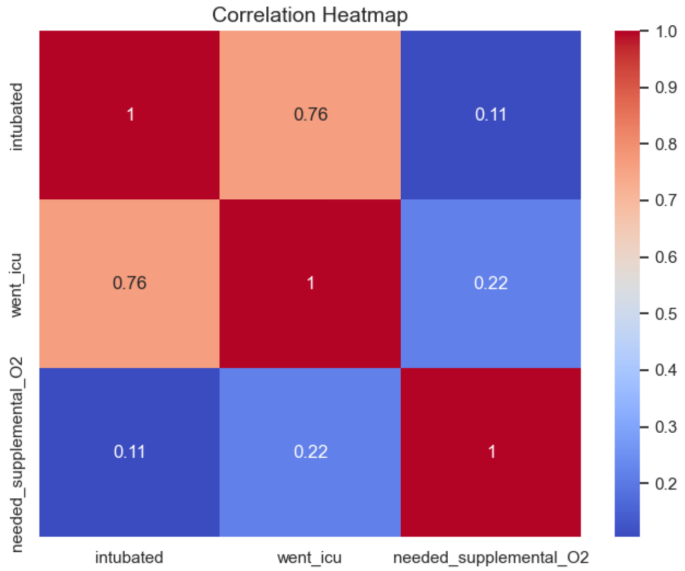


Figure 1.) Correlation Heatmap between some elements of the metadata that are correlated with a positive Covid-19 test.

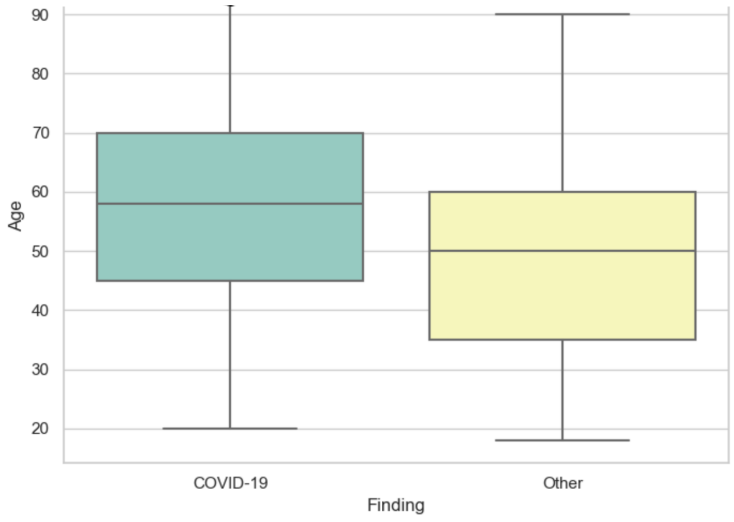


Figure 2.) Boxplot relationship between age and finding for entries in the metadata.

Figure 3.) Standard template example of a negative Chest X-Ray. No image occlusion in the lung area of the figure

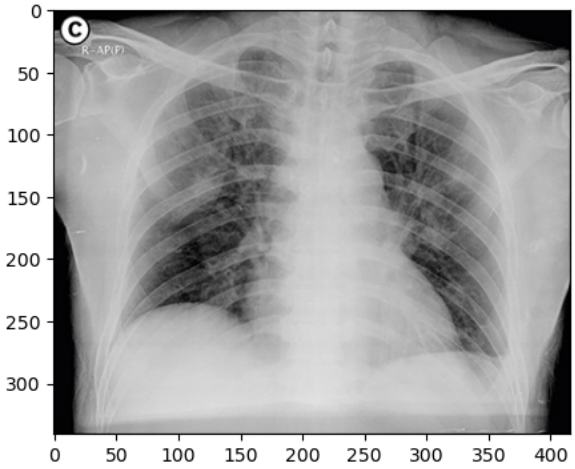


Figure 4.) Standard template example of a positive Covid-19 Chest X-Ray. Characterized by large amounts of image occlusion in the central cavity of the lungs, and mucus throughout the lung space.

Summary of Models

The dataset consists of 1650 labeled X-Ray images (585 positive, 1065 negative) and was preprocessed to include only PA images, which present the chest in anatomical position, resized to 256x256, and centered on the chest (area of interest). The data was split into training, validation, and testing sets.

Two models were trained to classify the X-Ray images as either positive or negative for pneumonia. The first model was a Sequential model with a simple architecture consisting of single convolutional layers of constantly increasing size from 16, 32, 64, 128, 256 with Average Pooling layers between each. The model then used a flatten, dense(512), and dense(1) connected layer structure to return a sigmoid 1 or 0 for positive or negative. However, this model did not perform well during validation and testing due to its simple architecture and inability to capture the nuanced features of the images.

The second model, which was drawn on the mistakes of the first layer, was designed to better capture the features of the data. This model utilized a smaller initial number of filters (16) to capture more nuanced components of each image and used Max pooling between layers. The model consisted of a 2x64 stack, a 2x128 stack, followed by a dense(256), dropout, and dense(1) connected layer. This denser neural network allowed the network to capture more nuanced features of the data, along with more general features given the increase of model parameters and complexity. Moreover, this model was trained on cleaner data, augmented data, and utilized dropout regularization, and Max pooling, which helped identify more generalized features of each X-Ray image.

Based on the training and validation accuracy, the second model performs significantly better than the first model, with a validation accuracy of +90% and test accuracy of ~70%, indicating that this model is capable of accurately classifying X-Ray images as positive or negative for Covid-19.

Below are some images generated from the creation of both models, along with some predictions and features of the more accurate model.

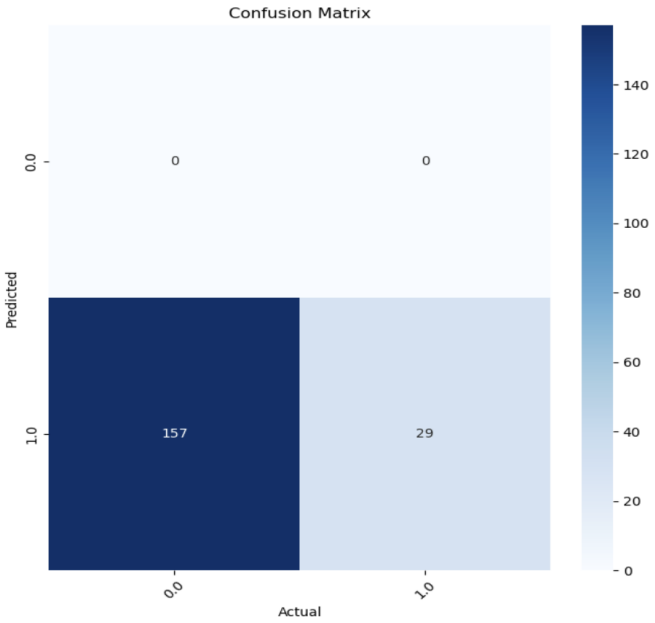


Figure 5.) Confusion matrix for validation data of model 1. The model was not able to differentiate between positive and negative images, and instead minimized loss by emulating the percentage of positive and negative images in training when making predictions.

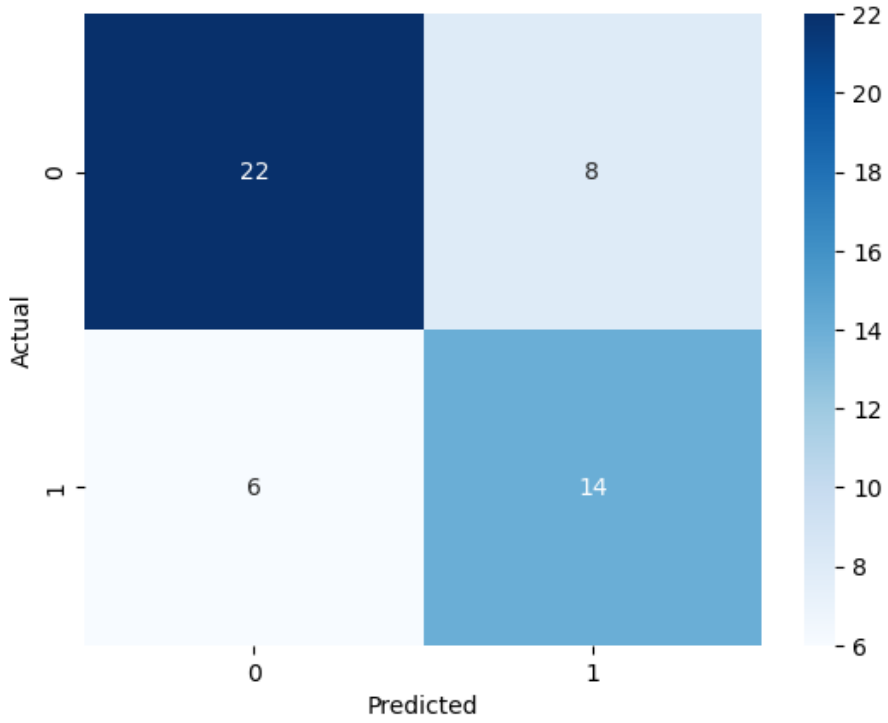


Figure 6.) Confusion matrix for testing data of model 2. The model was able to differentiate somewhat successfully between positive and negative images and is able to predict both classes with high and low degrees of confidence. (72% accuracy)

```
Epoch 1/10
8/8 [==============================] - 18s 2s/step - loss: 0.6860 - accuracy: 0.5117 - val_loss:
0.6729 - val_accuracy: 0.4844

Epoch 2/10
8/8 [==============================] - 17s 2s/step - loss: 0.6445 - accuracy: 0.5586 - val_loss:
0.6003 - val_accuracy: 0.4844

Epoch 3/10
8/8 [==============================] - 17s 2s/step - loss: 0.5249 - accuracy: 0.6562 - val_loss:
0.4396 - val_accuracy: 0.9062

Epoch 4/10
8/8 [==============================] - 15s 2s/step - loss: 0.3784 - accuracy: 0.8547 - val_loss:
0.3249 - val_accuracy: 0.8906

Epoch 5/10
8/8 [==============================] - 18s 2s/step - loss: 0.2884 - accuracy: 0.8828 - val_loss:
0.4788 - val_accuracy: 0.7656

Epoch 6/10
8/8 [==============================] - 16s 2s/step - loss: 0.3473 - accuracy: 0.8333 - val_loss:
0.2960 - val_accuracy: 0.8750

Epoch 7/10
8/8 [==============================] - 16s 2s/step - loss: 0.2923 - accuracy: 0.8974 - val_loss:
0.1588 - val_accuracy: 0.9531

Epoch 8/10
8/8 [==============================] - 17s 2s/step - loss: 0.2434 - accuracy: 0.9336 - val_loss:
0.3186 - val_accuracy: 0.9375

Epoch 9/10
8/8 [==============================] - 17s 2s/step - loss: 0.2875 - accuracy: 0.9219 - val_loss:
0.2494 - val_accuracy: 0.8906

Epoch 10/10
8/8 [==============================] - 16s 2s/step - loss: 0.1874 - accuracy: 0.9274 - val_loss:
0.0878 - val_accuracy: 0.9531
```

Figure 7.) Training history of model 2 with validation loss and accuracy included for clarity. The model is able to fit the cleaner data much better than model 1 and minimizes loss much better.

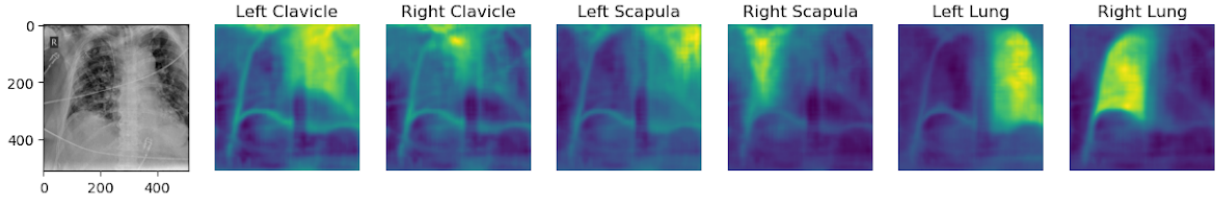
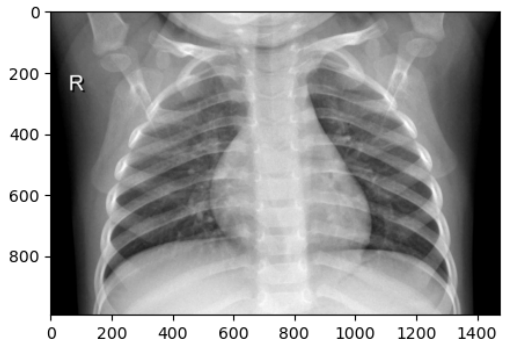


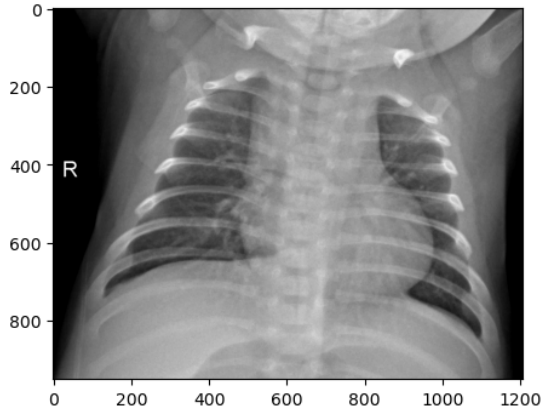
Figure 8.) Using keras for model and image classification allows the model to segment certain portions of the image during evaluation. Above is a demonstration of pre-built segmentation modules being used on a training image to highlight areas of interest.

```
Predicted Label:  NEGATIVE with probability: [1.5976102e-05] %
Actual Label:  NEGATIVE
```


```
Predicted Label:  POSITIVE with probability: [97.97355] %
Actual Label:  NEGATIVE
```




```
Predicted Label:  NEGATIVE with probability: [0.2727783] %
Actual Label:  NEGATIVE
```



```
Predicted Label:  POSITIVE with probability: [99.991455] %
Actual Label:  POSITIVE
```

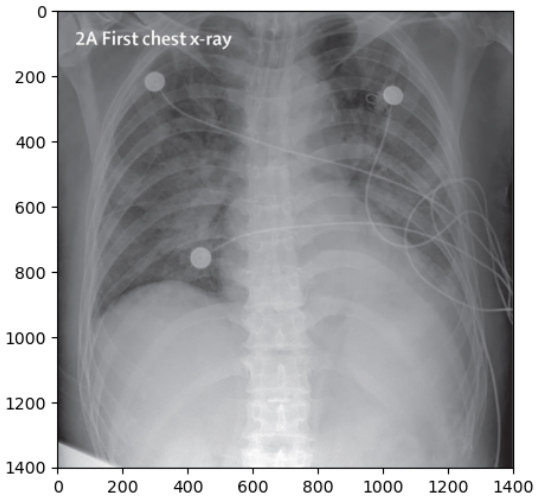


Figure 9.) Predictions for test images from model 2. The model can make predictions with a high degree of confidence, (1,4). However, the model may still be incorrect (2) or unsure (3).

Model: "sequential_10"

Layer (type)	Output Shape	Param #
conv2d_59 (Conv2D)	(None, 254, 254, 16)	448
max_pooling2d_56 (MaxPooling2D)	(None, 127, 127, 16)	0
conv2d_60 (Conv2D)	(None, 125, 125, 64)	9280
max_pooling2d_57 (MaxPooling2D)	(None, 62, 62, 64)	0
conv2d_61 (Conv2D)	(None, 60, 60, 64)	36928
max_pooling2d_58 (MaxPooling2D)	(None, 30, 30, 64)	0
conv2d_62 (Conv2D)	(None, 28, 28, 128)	73856
max_pooling2d_59 (MaxPooling2D)	(None, 14, 14, 128)	0
conv2d_63 (Conv2D)	(None, 12, 12, 128)	147584
max_pooling2d_60 (MaxPooling2D)	(None, 6, 6, 128)	0
flatten_7 (Flatten)	(None, 4608)	0
dense_14 (Dense)	(None, 256)	1179904
dropout_7 (Dropout)	(None, 256)	0
dense_15 (Dense)	(None, 1)	257

Total params: 1,448,257
Trainable params: 1,448,257
Non-trainable params: 0

Figure 10.) Final model architecture for image classification taken from model.summary().

Summary and Conclusion

Based on this project, I have learned that convolutional neural networks are powerful tools for image classification tasks. By applying image augmentation techniques, dropout regularization, and proper model architecture, it is possible to achieve high accuracy in identifying Covid-19 from chest X-ray images. However, it is important to have enough quality data to train the model, and increasing the size and diversity of the dataset would improve the model's performance.

Domain experts in the field could learn from this project by understanding the features that the model identifies in Covid-19 positive X-ray images. They could utilize this knowledge to better guide their diagnoses and treatment decisions, in combination with other diagnostic tools and lab data.

If I had more time and resources, I would gather more X-ray images, including images from other types of chest-occluding diseases. I would also introduce more complex regularization techniques to the model to further improve its generalization ability. Additionally, I would explore other deep learning architectures, such as attention-based models, to see if they could improve the accuracy of the classification task.