Covid-19 Identification from Chest X-Ray with a study on Google Search Correctness estimation

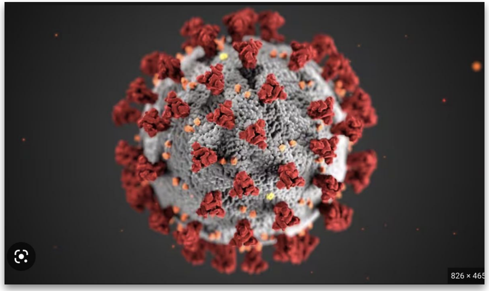
Sibashis Chatterjee
sibashis1992@gmail.com

Submitted as part of final project presentation of IISC CCE course AL & ML with Python

Motivation

- Covid-19 is a global pandemic affecting entire human race.
- Major Challenges in initial days
    - Unavailability of specialized test kits
    - Not enough PPEs
    - Not enough trained personnel

- Solution Exploration
    - Abundantly available radiology equipments (e.g. X-Ray)
    - Use of Machine Learning in attempt of classifying chest X-Ray as COVID affected
    - Use of CNN for image classification



Data Sources

- Using Google image search to download Covid 19 affected Lungs' X-Ray images

- Some data published by some universities:
  - The researchers of Qatar University have compiled the COVID-QU-Ex dataset, which consists of 33,920 chest X-ray (CXR) images including: 11,956 COVID-19, 11,263 Non-COVID infections (Viral or Bacterial Pneumonia), and 10,701 Normal. This data can be found here
  - University of Montreal released images can be found here

- Combining all these data to create a huge labeled set of Chest X-Ray dataset. I'll select training, cross validation and test data dynamically.

Technical setup

- Python (v >= 3.11)
- Latest PIP
- Libraries
    - google-api-python-client
    - numpy
    - pandas
    - matplotlib
    - PIL
    - tensorflow

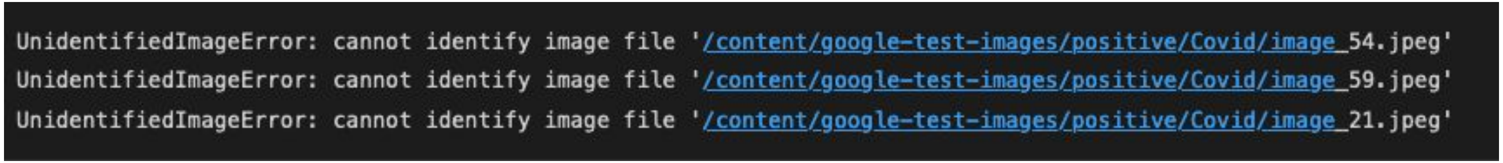
Data Collection

- Scraping Images from Google, refer: image_scrapping.ipynb.
- Using Google Custom Search API for the scraping. API Keys and other secrets are stored in a file ***my_secrets.py*** (Not pushed to remote repository for security reasons).
- The Scrapping worked till 200 images but then started returning **400 Bad Request** errors; probably I reached the free use limit.
- Download and collate data at one place for creating Training data set, refer classifier.ipynb.

Data cleaning

- Google returned a lot of images with wrong encoding and format information. This was causing model predictions to fail.

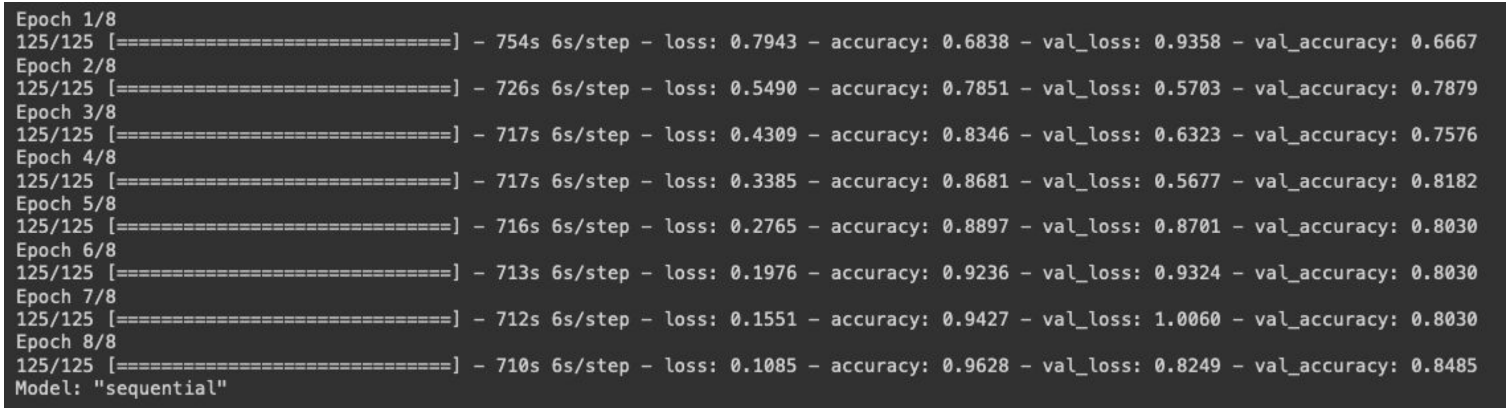
- Used Image package from PIL library to read the images and clean them up.



```
UnidentifiedImageError: cannot identify image file '/content/google-test-images/positive/Covid/image_54.jpeg'
UnidentifiedImageError: cannot identify image file '/content/google-test-images/positive/Covid/image_59.jpeg'
UnidentifiedImageError: cannot identify image file '/content/google-test-images/positive/Covid/image_21.jpeg'
```

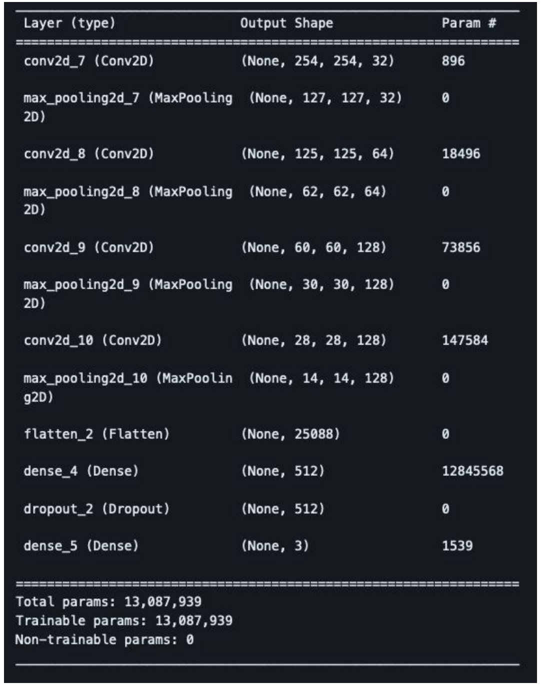
Model fitting

- Using Tersorflow Sequential model of Convolution Neural Network.
- By trial and error determined that a model with 4 hidden layers is performing best with some parameter adjustments.
- Refer classifier.ipynb for details about data preprocessing and training of the final model.
- I had tried with different number of layers and hyper parameters, one such example can be found at classifier1.ipynb and classifier_3_hidden_layers.ipynb files.



```
Epoch 1/8
125/125 [==============================] - 754s 6s/step - loss: 0.7943 - accuracy: 0.6838 - val_loss: 0.9358 - val_accuracy: 0.6667
Epoch 2/8
125/125 [==============================] - 726s 6s/step - loss: 0.5490 - accuracy: 0.7851 - val_loss: 0.5703 - val_accuracy: 0.7879
Epoch 3/8
125/125 [==============================] - 717s 6s/step - loss: 0.4309 - accuracy: 0.8346 - val_loss: 0.6323 - val_accuracy: 0.7576
Epoch 4/8
125/125 [==============================] - 717s 6s/step - loss: 0.3385 - accuracy: 0.8681 - val_loss: 0.5677 - val_accuracy: 0.8182
Epoch 5/8
125/125 [==============================] - 716s 6s/step - loss: 0.2765 - accuracy: 0.8897 - val_loss: 0.8701 - val_accuracy: 0.8030
Epoch 6/8
125/125 [==============================] - 713s 6s/step - loss: 0.1976 - accuracy: 0.9236 - val_loss: 0.9324 - val_accuracy: 0.8030
Epoch 7/8
125/125 [==============================] - 712s 6s/step - loss: 0.1551 - accuracy: 0.9427 - val_loss: 1.0060 - val_accuracy: 0.8030
Epoch 8/8
125/125 [==============================] - 710s 6s/step - loss: 0.1085 - accuracy: 0.9628 - val_loss: 0.8249 - val_accuracy: 0.8485
Model: "sequential"
```

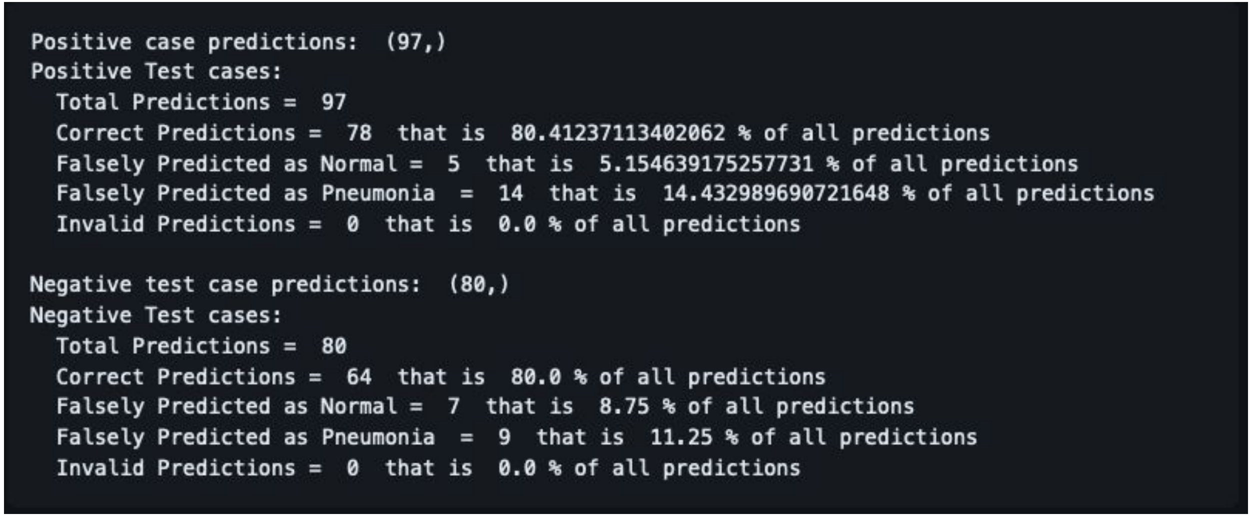
Model Summary



```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_7 (Conv2D)           (None, 254, 254, 32)      896

 max_pooling2d_7 (MaxPooling  (None, 127, 127, 32)     0
 2D)

 conv2d_8 (Conv2D)           (None, 125, 125, 64)      18496

 max_pooling2d_8 (MaxPooling  (None, 62, 62, 64)       0
 2D)

 conv2d_9 (Conv2D)           (None, 60, 60, 128)       73856

 max_pooling2d_9 (MaxPooling  (None, 30, 30, 128)      0
 2D)

 conv2d_10 (Conv2D)          (None, 28, 28, 128)       147584

 max_pooling2d_10 (MaxPoolin  (None, 14, 14, 128)      0
 g2D)

 flatten_2 (Flatten)         (None, 25088)             0

 dense_4 (Dense)             (None, 512)               12845568

 dropout_2 (Dropout)         (None, 512)               0

 dense_5 (Dense)             (None, 3)                 1539

=================================================================
Total params: 13,087,939
Trainable params: 13,087,939
Non-trainable params: 0
_________________________________________________________________
```

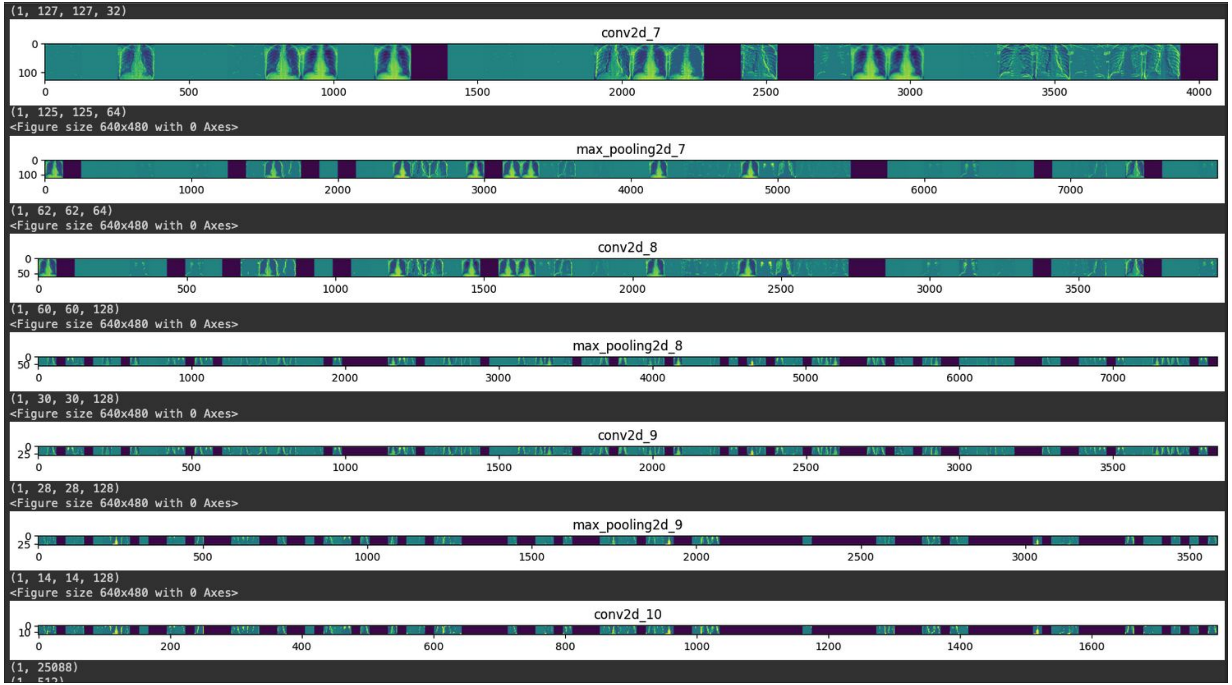
Analysis of Google Search Images

- I downloaded images from Google by searching with search phrase ***covid 19 infected lungs x ray***
- Downloaded images have Chest X-ray data, all of them expected to be classified with Covid-19.
- Analysis of Google Search results can be found at google_search_analysis.ipynb file.



```
Positive case predictions:  (97,)
Positive Test cases:
  Total Predictions =  97
  Correct Predictions =  78  that is  80.41237113402062 % of all predictions
  Falsely Predicted as Normal =  5  that is  5.154639175257731 % of all predictions
  Falsely Predicted as Pneumonia  =  14  that is  14.432989690721648 % of all predictions
  Invalid Predictions =  0  that is  0.0 % of all predictions

Negative test case predictions:  (80,)
Negative Test cases:
  Total Predictions =  80
  Correct Predictions =  64  that is  80.0 % of all predictions
  Falsely Predicted as Normal =  7  that is  8.75 % of all predictions
  Falsely Predicted as Pneumonia  =  9  that is  11.25 % of all predictions
  Invalid Predictions =  0  that is  0.0 % of all predictions
```

Features



```
(1, 127, 127, 32)
(1, 125, 125, 64)
<Figure size 640x480 with 0 Axes>
(1, 62, 62, 64)
<Figure size 640x480 with 0 Axes>
(1, 60, 60, 128)
<Figure size 640x480 with 0 Axes>
(1, 30, 30, 128)
<Figure size 640x480 with 0 Axes>
(1, 28, 28, 128)
<Figure size 640x480 with 0 Axes>
(1, 14, 14, 128)
<Figure size 640x480 with 0 Axes>
(1, 25088)
(1, 512)
```

Thank You