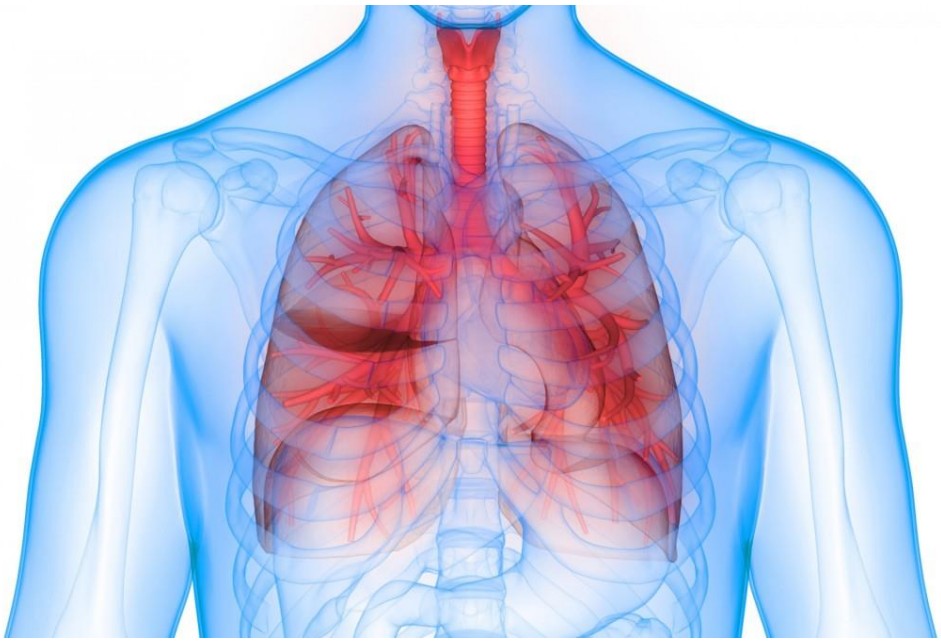
PNEUMONIA DETECTION Assignment

Final Report



Team:

Name	E-mail
Arup Kumar Mohanty	Arup.kumar1988@gmail.com
Mohit Gupta	Mohitgupta.mait@gmail.com
Ved Verma	Ved.verma.273306@gmail.com
Venkatesh Doddihal	Ds.venky1975@gmail.com

1 Table of Contents

1. Problem statement, Data, and findings

1.1 Understanding the Problem Statement

1.1.1 What is Pneumonia?

Pneumonia is an infection that inflames the air sacs in one or both lungs. The air sacs may fill with fluid or pus (purulent material), causing cough with phlegm or pus, fever, chills, and difficulty breathing. A variety of organisms, including bacteria, viruses, and fungi, can cause pneumonia.

Pneumonia can range in seriousness from mild to life threatening. It is most serious for infants and young children, people older than age 65, and people with health problems or weakened immune systems.

Most pneumonia occurs when a breakdown in your body's natural defences allows germs to invade and multiply within your lungs. To destroy the attacking organisms, white blood cells rapidly accumulate. Along with bacteria and fungi, they fill the air sacs within your lungs (alveoli). Breathing may be laboured. A classic sign of bacterial pneumonia is a cough that produces thick, blood-tinged, or yellowish-greenish sputum with pus.

It requires review of a chest radiograph by highly trained specialists. Pneumonia shows up in a chest radiograph as an area of opacity. However, diagnosis of it can be complicated and specialists in reviewing them spend much time and effort.

1.1.2 How dangerous is Pneumonia?

The UNICEF, in its own assessment, has put India in the second rank in terms of the number of deaths caused by pneumonia.

Most deaths occurred among children under the age of two. India accounts for 1.27 lakh deaths owing to pneumonia, ahead of Pakistan and Bangladesh. Only Nigeria is ahead of our country with 1.62 lakh such deaths.

Globally, the disease accounts for eight lakh deaths, one child every 39 seconds and just five countries are responsible for more than half of child pneumonia deaths — Nigeria (1,62,000), India (1,27,000), Pakistan (58,000), Democratic Republic of Congo (40,000) and Ethiopia (32,000). The UNICEF, in its own assessment, has put India in the second rank in terms of the number of deaths caused by pneumonia.

Most deaths occurred among children under the age of two. India accounts for 1.27 lakh deaths owing to pneumonia, ahead of Pakistan and Bangladesh. Only Nigeria is ahead of our country with 1.62 lakh such deaths.

Globally, the disease accounts for 8 lakh deaths, one child every 39 seconds and just five countries are responsible for more than half of child pneumonia deaths — Nigeria (1,62,000), India (1,27,000), Pakistan (58,000), Democratic Republic of Congo (40,000) and Ethiopia (32,000).

1.1.3 Is this treatable/preventable?

The disease can be prevented with vaccines and easily treated with low-cost antibiotics if accurately diagnosed. However, tens of millions of children are still going unvaccinated – and one in three with symptoms do not receive essential medical care.

In some severe cases the below image-guided treatments can be used for pneumonia:

1. Thoracentesis: Fluid may be taken from the chest cavity and studied to help the doctor determine which germ is causing your illness. X-ray, CT and/or ultrasound may be used during thoracentesis. The fluid removed during this procedure may also help provide symptom relief.

2. Chest tube placement: During this procedure, also known as thoracostomy, a thin plastic tube is inserted into the pleural space (the area between the chest wall and lungs. The tube can help remove excess fluid or air. The procedure is performed under the guidance of CT or ultrasound.

3. Image-guided abscess drainage: Image-guidance helps direct placement of a needle into the abscess cavity and can aid during insertion of a drainage tube. If an abscess has formed in the lungs, it may be drained by inserting a small drainage tube (catheter). Image guidance, including fluoroscopy, x-ray, ultrasound, or CT, is used.

1.1.4 So how to diagnose it?

While examination of symptoms is done to understand if the patient does indeed have pneumonia, these steps are usually preliminary steps like:

1. Understanding medical history and symptoms
2. Listening to the lungs - In checking for pneumonia, the doctor will listen for abnormal sounds like crackling, rumbling, or wheezing.

But if the doctor thinks that pneumonia is present, an imaging test may be performed to confirm the diagnosis.

One or more of the following tests may be ordered to evaluate for pneumonia:

1. Chest x-ray: An x-ray exam will allow the doctor to see the lungs, heart, and blood vessels to help determine if pneumonia is present. When interpreting the x-ray, the radiologist will look for white spots in the lungs (called infiltrates) that identify an infection. This exam will also help determine if there are any complications related to pneumonia such as abscesses or pleural effusions (fluid surrounding the lungs).
2. CT of the lungs: A CT scan of the chest may be done to see finer details within the lungs and detect pneumonia that may be more difficult to see on a plain x-ray. A CT scan also shows the airway (trachea and bronchi) in detail and can help determine if pneumonia may be related to a problem within the airway. A CT scan can also show complications of pneumonia, abscesses or pleural effusions and enlarged lymph nodes.

3. Ultrasound of the chest: Ultrasound may be used if fluid surrounding the lungs is suspected. An ultrasound exam will help determine how much fluid is present and can aid in determining the cause of the fluid.

4. MRI of the chest: MRI is not used to evaluate for pneumonia but may be used to look at the heart, vessels of the chest and chest wall structures. If the lungs are abnormal because of excess fluid, infection or tumour, an MRI may provide additional information about the cause or extent of these abnormalities.

5. Needle biopsy of the lung: The doctor may request a biopsy of your lung(s) to determine the cause of pneumonia. This procedure involves removing several small samples from your lung(s) and examining them. Biopsies of the lung can be done using x-ray, CT, ultrasound and/or MRI.

Chest X-rays are the commonly used method to detect the Pneumonia infection and locate the infected area in the lungs. In addition, the chest X-ray is the widely used radiological examination technique toward diagnosis of several lung diseases. Finding radiological examiners in remote places for analysis for a greater number of Chest X-rays is an extremely challenging task. In recent times, artificial intelligence approaches are used to solve the challenges in several of medical diagnosis processes.

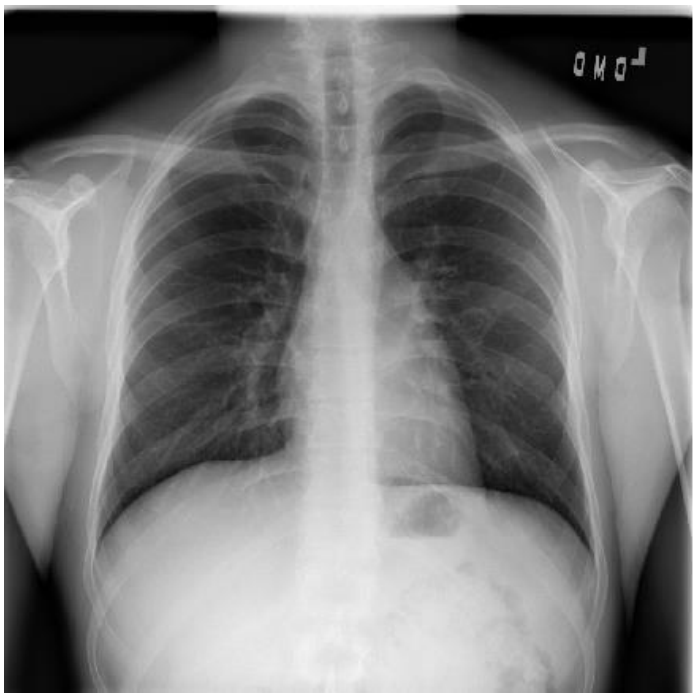
1.1.5 Chest Radiographs basics

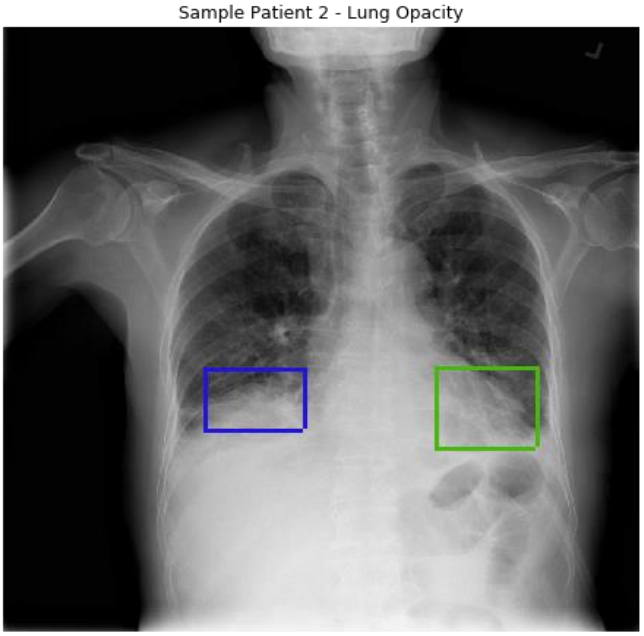
In the process of taking an image, an X-ray passes through the body and reaches a detector on the other side. Tissues with sparse material, such as lungs, which are full of air, do not absorb X-rays and appear black in the image. Dense tissues such as bones absorb X-rays and appear white in the image. In short –

* Black = Air
* White = Bone
* Grey = Tissue or Fluid

The left side of the subject is on the right side of the screen by convention. You can also see the small L at the top of the right corner. We see the lungs as black in a normal image, but they have different projections on them — the rib cage bones, main airways, blood vessels and the heart.

An example chest radiograph looks like this:





However, the diagnosis of pneumonia on CXR is complicated because of a number of other conditions in the lungs such as fluid overload (pulmonary edema), bleeding, volume loss (atelectasis or collapse), lung cancer, or post-radiation or surgical

changes. Outside of the lungs, fluid in the pleural space (pleural effusion) also appears as increased opacity on CXR. When available, comparison of CXRs of the patient taken at different time points and correlation with clinical symptoms and history are helpful in making the diagnosis.

In addition, clinicians are faced with reading high volumes of images every shift. Being tired or distracted clinicians can miss vital details in image.

Chest radiograph is the most performed diagnostic imaging study. Due to the high volume of chest radiography, it is very time consuming and intensive for the radiologists to review each image manually.

As such, an automated solution is ideal to locate the position of inflammation in an image. By having such an automated pneumonia screening system, this can assist physicians to make better clinical decisions or even replace human judgement in this area.

Here automated image analysis tools can come to help. For example, **one can use machine learning to automate initial detection (imaging screening) of potential pneumonia cases in order to prioritize and expedite their review**. Hence, we decided to develop a model to detect pneumonia from chest radiographs.

1.2 Project Objectives:

The objective is to build an algorithm that can detect visual signals for pneumonia in medical images.

Specifically, the algorithm needs to automatically locate lung opacities on chest radiographs, but only the opacities that look like pneumonia, and discard other types of opacities like the ones caused by

- fluid overload (pulmonary edema)
- bleeding, volume loss (atelectasis or collapse)
- lung cancer
- post-radiation
- surgical changes etc.

Outside of the lungs, fluid in the pleural space (pleural effusion) also appears as increased opacity on CXR. A pneumonia opacity is a part of the lungs that looks darker on a radiograph and has a shape that indicates that pneumonia is (or may be) present.

As the objective is to detect and draw a bounding box on each of the pneumonia opacities, where each image can have zero or many opacities, and the training set is already classified, it can be analysed as a supervised learning statistical multi label classification.

- Build a deep learning a pneumonia detection system, to locate the position of inflammation in an image.

- Use TensorFlow / Keras as the framework for building the model
- Read Medical images are stored in a special format called DICOM files (*.dcm).

1.3 Data & Findings:

- Details about the data and dataset files are given in below link,

  https://www.kaggle.com/c/rsna-pneumonia-detection-challenge/data

- The first step would be to examine the data available for this. The data is given in a zip file "rsna-pneumonia-detection-challenge.zip", which contains the following items:

- A folder "stage_2_train_images": This folder contains all the training dataset chest radiograph DICOM images.

- A csv file "stage_2_train_labels.csv": This file contains the corresponding patientID images to the folder "stage_2_train_images" and contains the bounding box of areas of pneumonia detected in each image along with a target label of 0 or 1 for pneumonia detected.

- A csv file "stage_2_detailed_class_info.csv": This file contains the corresponding patientID images to the folder "stage_2_train_images" and contains the target class labels of the images.

- A folder "stage_2_test_images": This folder contains all the test dataset chest radiograph DICOM images. We will not be using this set of images, as they do not contain labels.
- A csv file "stage_2_sample_submission.csv": This file contains the corresponding patientID images to the folder "stage_2_test_images". We will not be using this set of file.

2. EDA

2.1 Approach:

- Support the building of a neural network, the project will be done on **google Colab**.
  - The first step is to unzip the zip file to open the above files to the google drive directory.
  - Second step is to verify the format of images as provided, and they are all DICOM images in the "dcm" file format. Read the DICOM images using the **pydicom** library for that purpose. After that, the next step would be to inspect the csv files.

2.2 Analysis:

- Loading the "**stage_2_detailed_class_info.csv**" file into pandas dataframe, a quick glance reveals that it has only 2 columns:

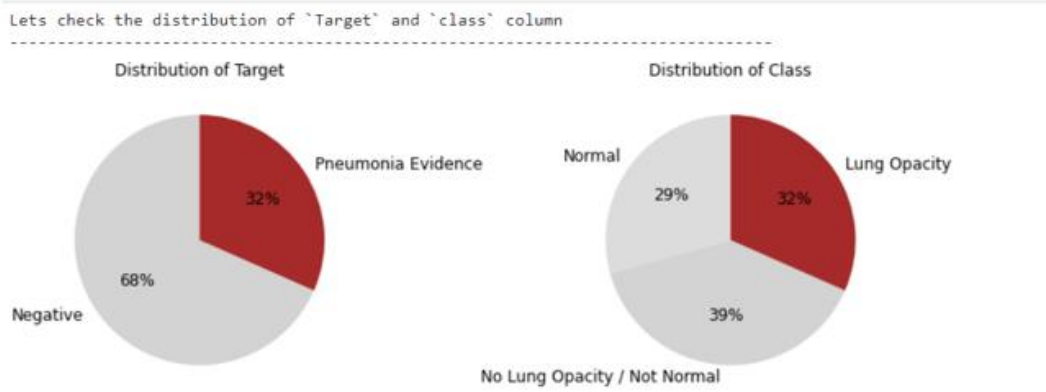
    - **patientId** – which refers to the patientId's corresponding image name

    - **class** – Target label of the patientId's image

```
# Check for types of class labels

df_class["class"].value_counts()
```

```
No Lung Opacity / Not Normal    11821
Lung Opacity                     9555
Normal                           8851
Name: class, dtype: int64
```



- As illustrate in above figure, there are 3 types of classes: **Normal, Lung Opacity and No Lung Opacity / Not Normal**. 11821(~39%) records belong to No Lung Opacity / Not Normal, 32% accounts for Lung Opacity and roughly 29% marked as Normal. The primary concern of the project would be to detect images with Lung Opacity, and the others would be in the same group labelling.

- The Target distribution seems to be imbalance as 32% of the patients are having pneumonia evidence whereas 68% are normal.

- Loading the "**stage_2_train_labels.csv**" file into panda's data frame, we can see that it has below fields:

    patientId – which refers to the patientId's corresponding image name
    x - upper-left x coordinate of the bounding box

y - upper-left y coordinate of the bounding box
width – the width of the bounding box
height – the height of the bounding box
Target – binary target indicating if this image has evidence of pneumonia

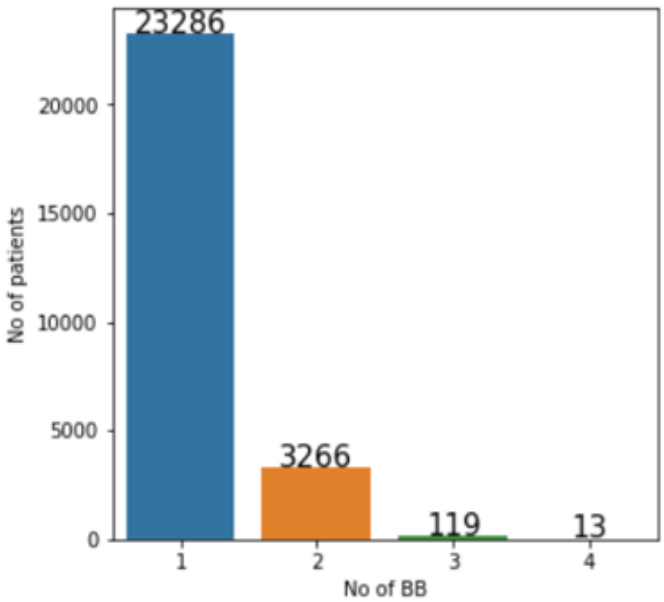
- There is total of 30,227 entries, no missing values with 9,555 images have bounding boxes. This corresponds to the data available "stage_2_detailed_class_info.csv" file.

```
train_labels_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 30227 entries, 0 to 30226
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   patientId   30227 non-null  object
 1   x           9555 non-null   float64
 2   y           9555 non-null   float64
 3   width       9555 non-null   float64
 4   height      9555 non-null   float64
 5   Target      30227 non-null  int64
dtypes: float64(4), int64(1), object(1)
memory usage: 1.4+ MB
```

The bounding box information is present for only 9555 records out of 30227.

number_of_boxes	number_of_patientIDs_per_boxes
1	23286
2	3266
3	119
4	13



- There are multiple records for patients. Number of duplicates in patientID is 3,543.
- After merging both the csv files below are the observations.
- About 23,286 patients (~87% of them) provided have 1 bounding box while 13 patients have 4 bounding boxes. The reason is that each row records a single

bounding box area of pneumonia detected. However, in a patient image, it might be the case of several bounding boxes area of pneumonia detected.

- Chest examinations with Target = 1 i.e., ones with evidence of Pneumonia are associated with Lung Opacity class.

- Chest examinations with Target = 0 i.e., those with no definitive evidence of Pneumonia are either of Normal or No Lung Opacity / Not Normal class.

- The next step is to read the images in the file "**stage_2_train_images**". Images provided are stored in DICOM (.dcm) format, which is an international standard to transmit, store, retrieve, print, process, and display medical imaging information. We will make use of pydicom package here to read the images.

```
Dataset.file_meta -------------------------------
(0002, 0000) File Meta Information Group Length  UL: 200
(0002, 0001) File Meta Information Version       OB: b'\x00\x01'
(0002, 0002) Media Storage SOP Class UID         UI: Secondary Capture Image Storage
(0002, 0003) Media Storage SOP Instance UID      UI: 1.2.276.0.7230010.3.1.4.8323329.6379.1517874325.469569
(0002, 0010) Transfer Syntax UID                 UI: JPEG Baseline (Process 1)
(0002, 0012) Implementation Class UID            UI: 1.2.276.0.7230010.3.0.3.6.0
(0002, 0013) Implementation Version Name         SH: 'OFFIS_DCMTK_360'
-------------------------------------------------
(0008, 0005) Specific Character Set              CS: 'ISO_IR 100'
(0008, 0016) SOP Class UID                       UI: Secondary Capture Image Storage
(0008, 0018) SOP Instance UID                    UI: 1.2.276.0.7230010.3.1.4.8323329.6379.1517874325.469569
(0008, 0020) Study Date                          DA: '19010101'
(0008, 0030) Study Time                          TM: '000000.00'
(0008, 0050) Accession Number                    SH: ''
(0008, 0060) Modality                            CS: 'CR'
(0008, 0064) Conversion Type                     CS: 'WSD'
(0008, 0090) Referring Physician's Name          PN: ''
(0008, 103e) Series Description                  LO: 'view: AP'
(0010, 0010) Patient's Name                      PN: '00436515-870c-4b36-a041-de91049b9ab4'
(0010, 0020) Patient ID                          LO: '00436515-870c-4b36-a041-de91049b9ab4'
(0010, 0030) Patient's Birth Date                DA: ''
(0010, 0040) Patient's Sex                       CS: 'F'
(0010, 1010) Patient's Age                       AS: '32'
(0018, 0015) Body Part Examined                  CS: 'CHEST'
(0018, 5101) View Position                       CS: 'AP'
(0020, 000d) Study Instance UID                  UI: 1.2.276.0.7230010.3.1.2.8323329.6379.1517874325.469568
(0020, 000e) Series Instance UID                 UI: 1.2.276.0.7230010.3.1.3.8323329.6379.1517874325.469567
(0020, 0010) Study ID                            SH: ''
(0020, 0011) Series Number                       IS: '1'
(0020, 0013) Instance Number                     IS: '1'
(0020, 0020) Patient Orientation                 CS: ''
```

- From the above sample we can see that dicom file contains some of the information that can be used for further analysis such as **sex, age, body part examined, view position and modality**. Size of this image is 1024 x 1024 (rows x columns).

- Examine further we will merge the image features with the existing class data. This will help us understand distribution of age for those with evidence of lung opacity and those with no definite evidence of lung opacity.

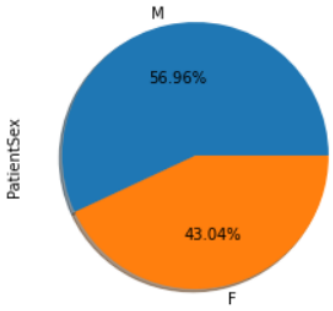
- Understand distribution of male and female for those with evidence of lung opacity and those with no definite evidence of lung opacity

- Explore different view positions in the dataset
- Explore modality
- We will pickle the file and do our analysis on the saved file

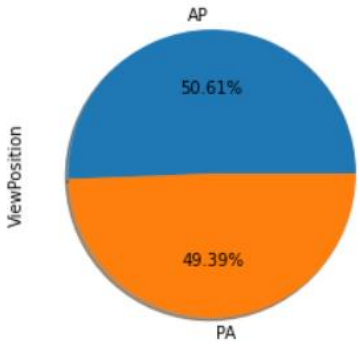
2.3 Visualization:

- As we proceed, further we will use different visualization techniques like univariate, multivariate analysis to discover patterns and anomalies in the data.

Univariate Analysis:

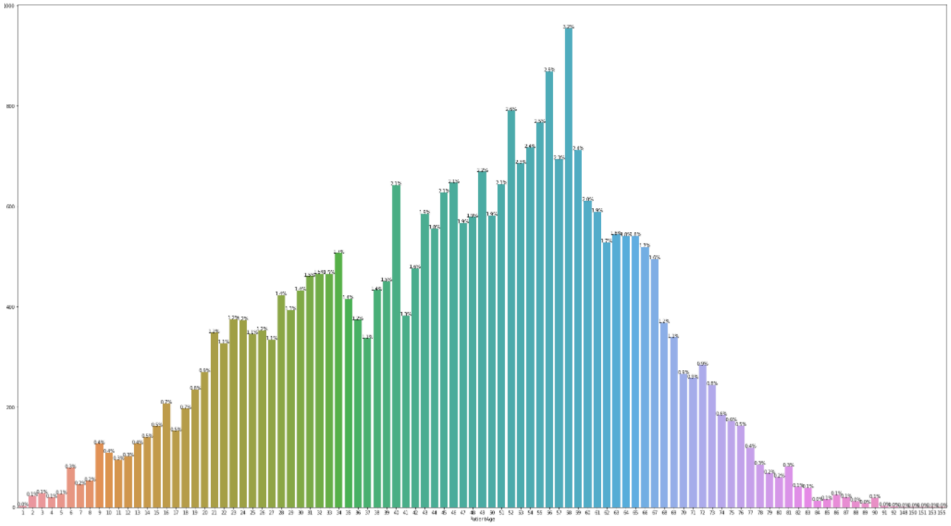


- As shown above, there are 56.96% male patients and 43% female patients. There are also more no of male patients having pneumonia compared to females.

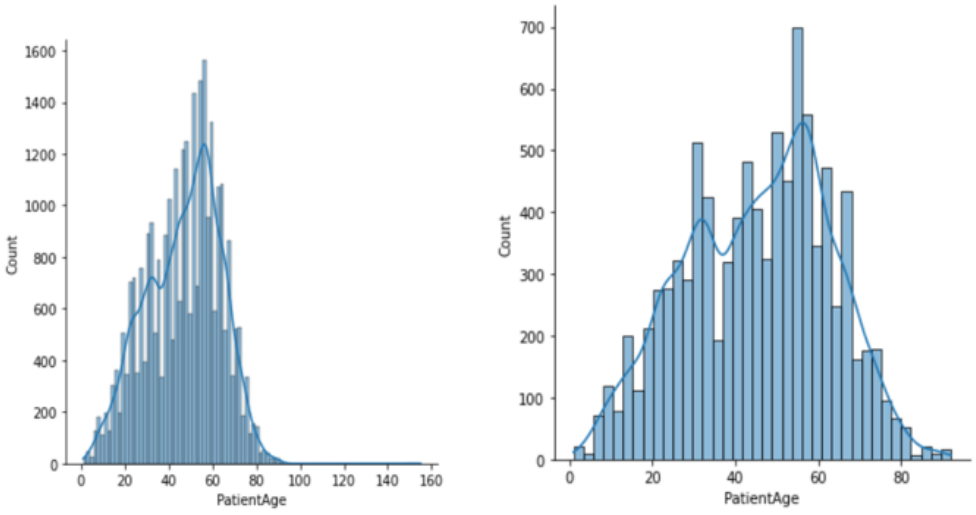


- We have two different view positions AP (Anterior/Posterior) and PA (Posterior/Anterior) in the training dataset.
- **Posterior/Anterior (PA)**: In PA, X-Ray beam hits the posterior (back) part of the chest before the anterior (front) part. While obtaining the image patient is asked to stand with their chest against the film.
- **Anterior/Posterior (AP)**: At times, it is not possible for radiographers to acquire a PA chest X-ray. This is usually because the patient is too unwell to stand. AP projection images are of lower quality than PA images. Heart size is exaggerated (cardiothoracic ratio approximately 50%)

- As can be seen above in the chart, the view position attributes are almost equally distributed.



- The above graphs shows the distribution of age across all the patients. As per the data the maximum no of patients are falling within age group 40-60.
- There are also few patients where age > 100. These records seems to be erronous and we can safely remove them from our dataset and will keep the age



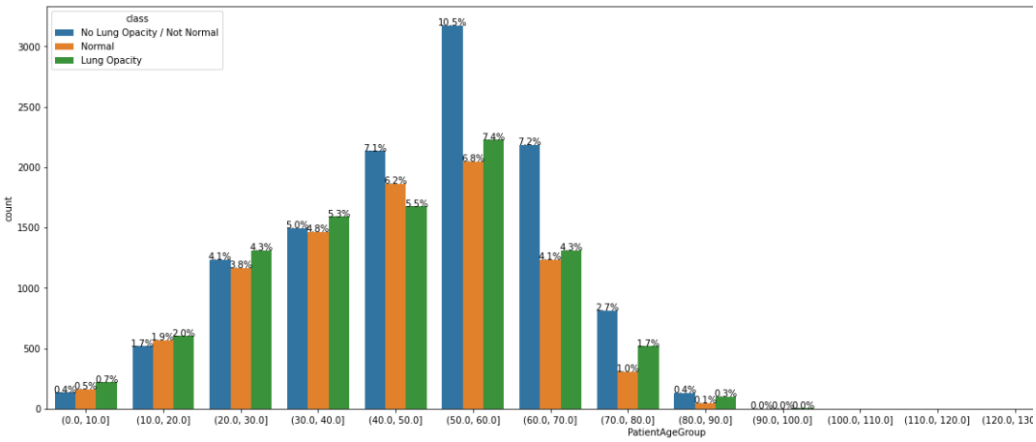
- As per the above histogram, the patient age is normally distributed with more volume of data lying between age group 40-60.
- The Age distribution for pnenumonia patient is slightly left skewed with more no of patient between age group 40-60 are having pneumonia.

Bivariate Analysis:

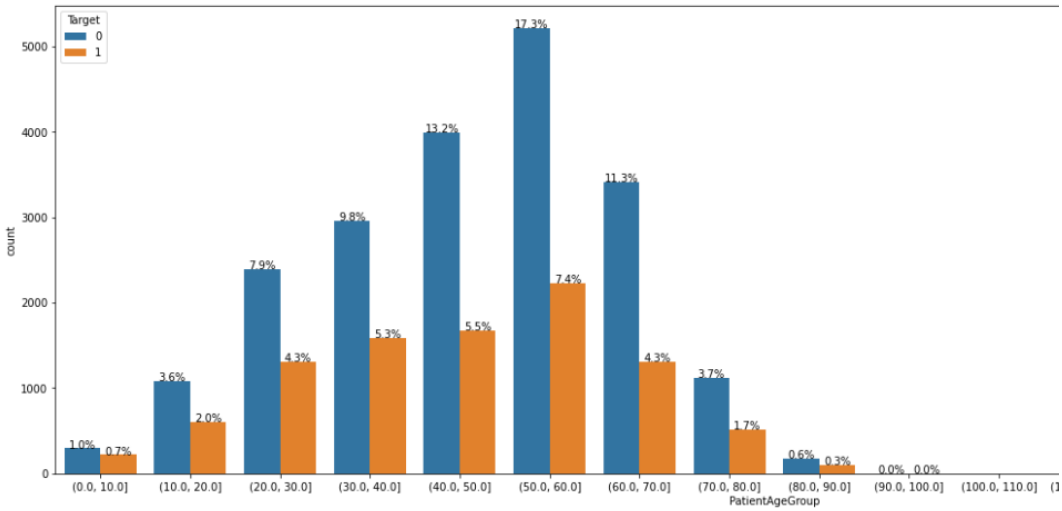
- To have a better interpretation we will make use of binning concept as shown below to group the patient age and will further visualize on the transformed data.

```
merged_df['PatientAgeGroup'] = pd.cut(merged_df['PatientAge'], patient_age_arr)
merged_df['PatientAgeGroup'].value_counts()
```

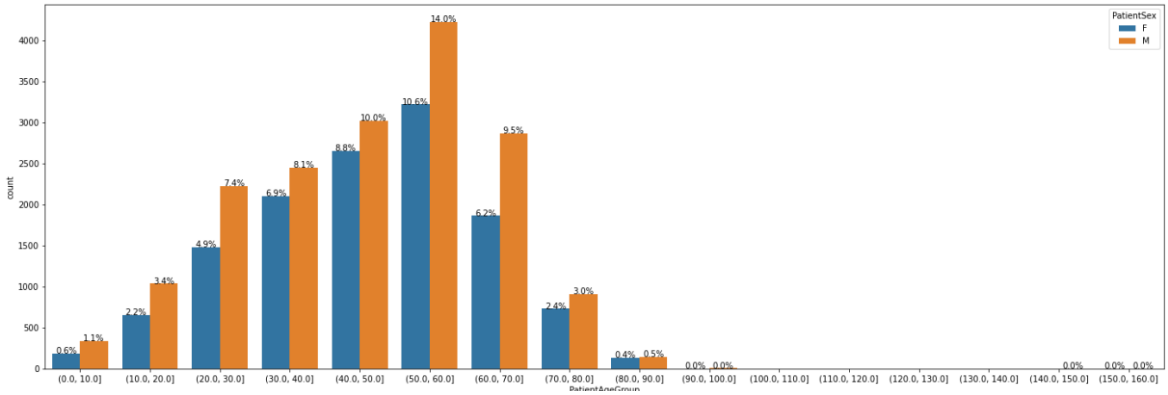
```
(50.0, 60.0]      7446
(40.0, 50.0]      5671
(60.0, 70.0]      4730
(30.0, 40.0]      4551
(20.0, 30.0]      3704
(10.0, 20.0]      1688
(70.0, 80.0]      1637
(0.0, 10.0]        515
(80.0, 90.0]       275
(90.0, 100.0]        5
(150.0, 160.0]       3
(140.0, 150.0]       2
(100.0, 110.0]       0
(110.0, 120.0]       0
(120.0, 130.0]       0
(130.0, 140.0]       0
Name: PatientAgeGroup, dtype: int64
```



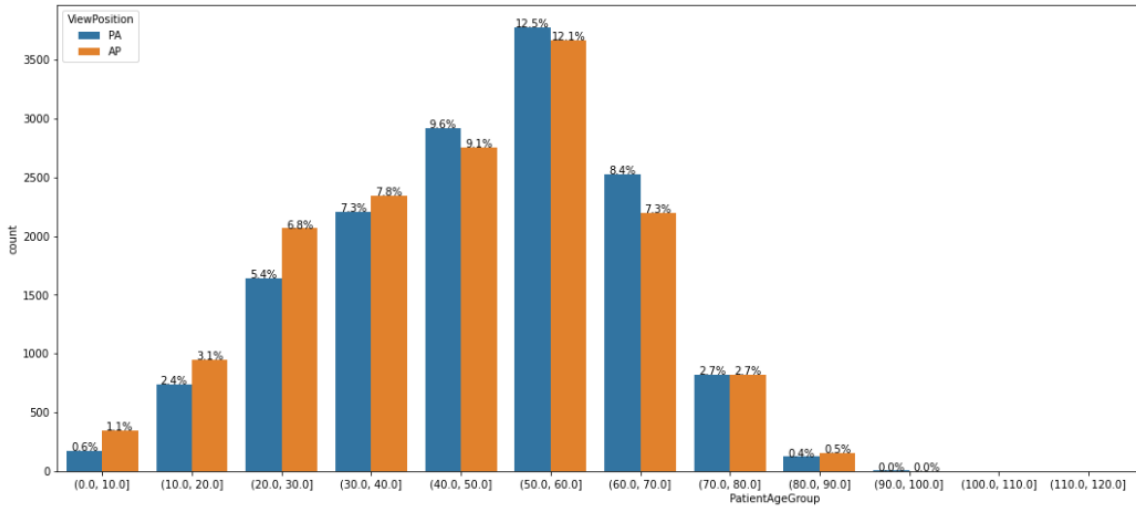
- The above picture depicts a distribution of different classes across the 'PatientAgeGroup'.
- As can be seen patients with age group between 50-60 are having the highest probability of getting pnenumonia compared to other age groups.
- There are very less no of data points for age group>70.



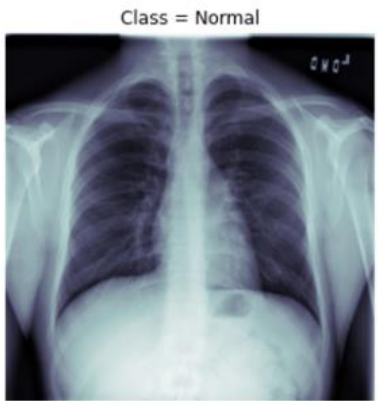
- The distribution of Age group with Target variable also reflects maximum no of postive cases between 40-60. For age group 50-60 there are 17.3% of patients who doesn't have pneumonia compared to 7.4% who have.

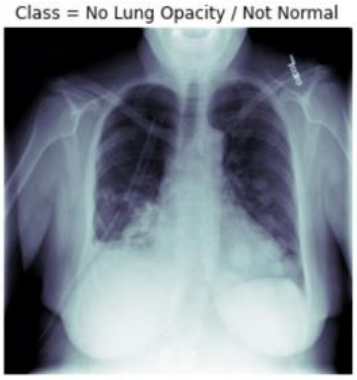


- The distribution of patient sex vs Age group shows there are maximum no of male and female present in age group 50-60 which is 14% and 10.6% respectively.
- Next majority falls between age group 40-50.
- There are very less no of records exists for age group 0-10 and 80-100.

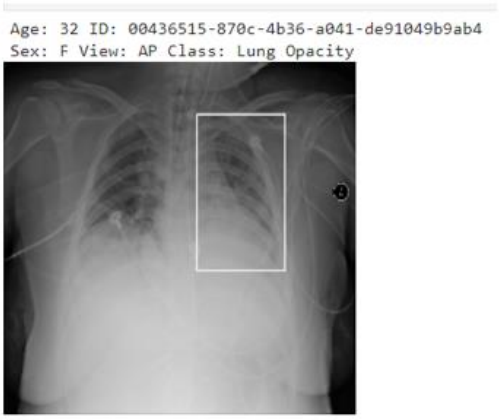


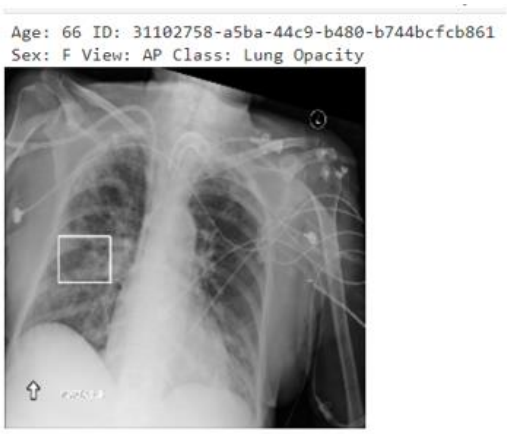
- Above picture illustrate the PA and AP view position for different Age group.
- Data shows for age group 50-60 there are 12.5% patients with AP and 12.1 % with PA position.
- Amongs all the patients between age group 30-40, 9.6% are having AP and 9.1% are having PA view positions.
- For agegroup 0-10 and 80-100 we have very less data available.
- Before proceeding, it would be good to view how the images are displayed. So loading a few DICOM images in google Colab. We will read images from training samples for both normal as well as for pneumonia patients as shown below.





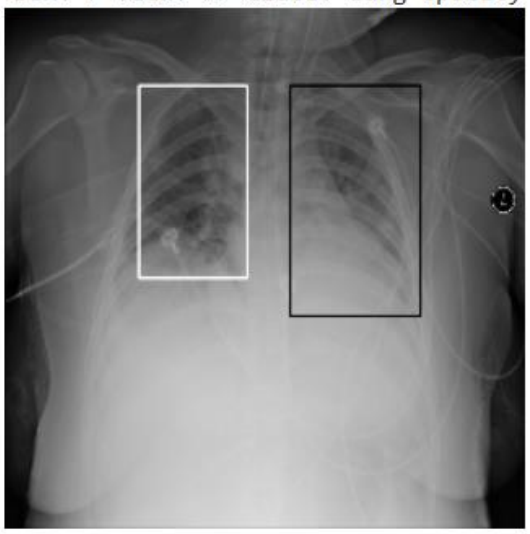
- The above two images represent classes **'Normal'** and '**No Lung Opacity / Not Normal.'** In the normal class, the image is quite prominent with no sign of any lung's opacity.
- The second image signifies a patient with **No Lung Opacity / Not Normal.'** Even though it does not have any Lung Opacity, still some portion of the image is blurred giving a notion of Not normal lungs. Medical practitioners will further validate this sort of cases.



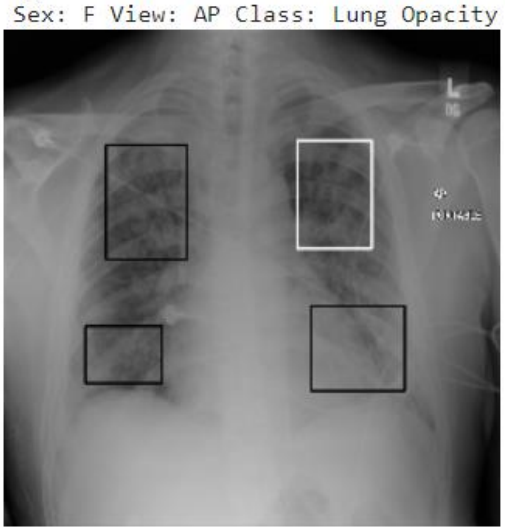


- Above two screenshot shows patients having lungs opacity. As can be seen there are bounding box on each of the images pointing the infections. There will be region of infection based on which the bounding box coordinates have been defined.

```
00436515-870c-4b36-a041-de91049b9ab4 2
00436515-870c-4b36-a041-de91049b9ab4 264.0 152.0 477.0 531.0
00436515-870c-4b36-a041-de91049b9ab4 562.0 152.0 818.0 605.0
Age: 32 ID: 00436515-870c-4b36-a041-de91049b9ab4
Sex: F View: AP Class: Lung Opacity
```



```
ee820aa5-4804-4984-97b3-f0a71d69702f 4
ee820aa5-4804-4984-97b3-f0a71d69702f 605.0 228.0 759.0 451.0
ee820aa5-4804-4984-97b3-f0a71d69702f 634.0 569.0 827.0 744.0
ee820aa5-4804-4984-97b3-f0a71d69702f 211.0 238.0 378.0 473.0
ee820aa5-4804-4984-97b3-f0a71d69702f 168.0 611.0 326.0 728.0
Age: 72 ID: c1f7889a-9ea9-4acb-b64c-b737c929599a
Sex: F View: AP Class: Lung Opacity
```



- In addition, there are patient id's which are duplicate or have multiple bounding boxes. This is valid as the patients might have multiple area with infections.
- As shown above the first image shows two bounding boxes/lungs opacity for the same patient id.
- Similarly, for the second image there are four regions of infection for the same patient.

2.4 Summary:

- The training dataset (both csv files and the training image folder) contains information of 26684 patients (unique)
- Out of these 26684 unique patients some of these have multiple entries in both of the csv files
- Most of the recorded patient belong to Target = 0 (i.e., they do not have Pneumonia)
- Some of the patients have more than one bounding box. The maximum being four
- The classes "No Lung Opacity / Not Normal" and "Normal" is associated with Target = 0 whereas "Lung Opacity" belong to Target = 1
- The images are present in dicom format, from which information like PatientAge, PatientSex, ViewPosition etc are obtained
- There are two ways from which images were obtained: AP and PA. The age ranges from 1-155 (which were further clipped to 100)
- The centres of the bounding box are spread out over the entire region of the lungs. However, there are some centres, which are outliers.

3. Pre-Processing

This section describes the pre-processing steps applied to data before modelling. The images are in dicom format, which contains lot of metadata along with pixel data. The pixel data needs to be extracted and converted to either jpg or png format.

3.1 Pre-processing Methods

The following pre-processing methods can be applied to images.

- Conversion of image to jpg or png format.
- Find the number of channels in images and align to 1 or 3 channels.
    - Convert images to grey scale.
- Image resizing required as per base model requirements like 224*224 for VGG16.
- Drop duplicate data.
- Set null values to 0 or drop the rows.
- Pixel normalization. The pixel intensity values are modified to a range of values This is also known as contrast stretching.
- Image augmentation is technique used to artificially generate more variations in existing data and create additional variety in training images. This technique helps in generalizing the model better and avoids overfitting. Some of the transformations that can be applied are,
    - Rotate the images by specified angle.
    - Flip vertically or horizontally.
    - Shearing - shift one part of the image like parallelogram.
    - Generate masks for the image.
    - Thresholding – used to binarize grey scale images.
    - Erosion, Dilation – Used to either erode certain features or make them prominent.
    - Crop the images and generate many sub-sets from original images.

3.2 Pre-processing Applied

- The data generators are used to pre-process the image.
- The images are resized to 224*224 and processed in batches of 32.
- Duplicate rows were dropped from merged data frame “train_feature_engineered”.
- The total number records after dropping duplicates are 26684.
- The distribution of target variable and classes are given below.

```
Distribution of target and classes
0    20672
1     6012
Name: Target, dtype: int64

No Lung Opacity / Not Normal    11821
Normal                           8851
```

```
Lung Opacity        6012
Name: class, dtype: int64
```

- The shape of data after split into train, test and validation are as below.

```
Shape of the dataframes:
TRAIN:(21348, 3)
VALID:(2668, 3)
TEST:(2668, 3)
```

- The data distribution is proper across train, test and validation data set.

```
Distribution of target in the training set:
0  0.78
1  0.22
Name: Target, dtype: float64

Distribution of target in the validation set:
0  0.78
1  0.22
Name: Target, dtype: float64

Distribution of target in the test set:
0  0.77
1  0.23
Name: Target, dtype: float64
```

4. Model and Model Building

This section describes the approach used in modelling and the different techniques applied along with their evaluation.

4.1 Models

The problem of lung opacity detection involves both classification and regressions. It is a classification problem, as the model needs to identify if lung opacity is present or absent to aid in the detection of pneumonia. It is a regression problem, as the model needs to identify the pixel areas containing the lung opacity. Some of the techniques, which can be used, are,

- Fast R-CNN
- Faster R-CNN
- Single Shot Detector (SSD)
- YOLO (You Look Only Once)
- SPP-net

Techniques based on R-CNN require lot of images for training and requires lot of training time with high resource consumption. SSD and YOLO perform better on time and resources comprising on accuracy. To overcome the above limitations, techniques of transfer learning based on existing models and their weights can be used to get better accuracy with reduced training data set in relatively less time.

Some of the pre-trained models suitable for opacity detection are,

- Densenet-121
- VGG 16
- ResNet-50
- CheXNet

4.1.1 DenseNet-121

DenseNet-121 has been designed to address the "vanishing gradient" problem with traditional CNN's as number of layers become more. In this architecture each layer is connected directly to every layer. In case of N Layers, there are N(N+1)/2 connections.

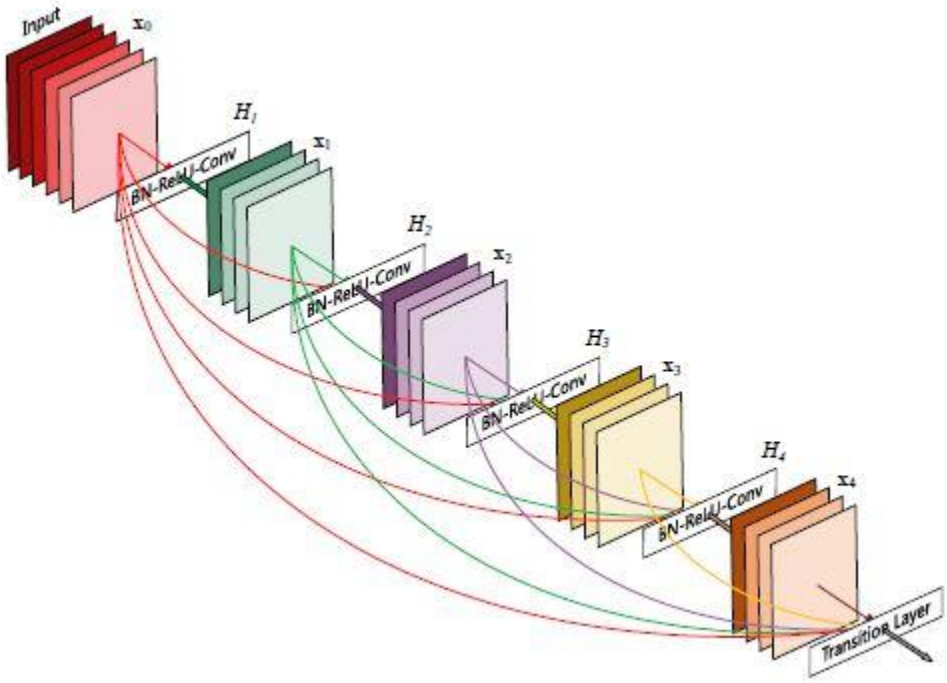


Figure 1: DenseNet-121 Architecture

The following layers are present in DenseNet-121,

- 1 7x7 Convolution
- 58 3x3 Convolution
- 61 1x1 Convolution
- 4 AvgPool
- 1 Fully Connected Layer

Layers	Output Size	DenseNet-121	DenseNet-169	DenseNet-201	DenseNet-264
Convolution	112 × 112	7 × 7 conv, stride 2			
Pooling	56 × 56	3 × 3 max pool, stride 2			
Dense Block (1)	56 × 56	[1 × 1 conv; 3 × 3 conv] × 6	[1 × 1 conv; 3 × 3 conv] × 6	[1 × 1 conv; 3 × 3 conv] × 6	[1 × 1 conv; 3 × 3 conv] × 6
Transition Layer (1)	56 × 56	1 × 1 conv			
	28 × 28	2 × 2 average pool, stride 2			
Dense Block (2)	28 × 28	[1 × 1 conv; 3 × 3 conv] × 12	[1 × 1 conv; 3 × 3 conv] × 12	[1 × 1 conv; 3 × 3 conv] × 12	[1 × 1 conv; 3 × 3 conv] × 12
Transition Layer (2)	28 × 28	1 × 1 conv			
	14 × 14	2 × 2 average pool, stride 2			
Dense Block (3)	14 × 14	[1 × 1 conv; 3 × 3 conv] × 24	[1 × 1 conv; 3 × 3 conv] × 32	[1 × 1 conv; 3 × 3 conv] × 48	[1 × 1 conv; 3 × 3 conv] × 64
Transition Layer (3)	14 × 14	1 × 1 conv			
	7 × 7	2 × 2 average pool, stride 2			
Dense Block (4)	7 × 7	[1 × 1 conv; 3 × 3 conv] × 16	[1 × 1 conv; 3 × 3 conv] × 32	[1 × 1 conv; 3 × 3 conv] × 32	[1 × 1 conv; 3 × 3 conv] × 48
Classification Layer	1 × 1	7 × 7 global average pool			
		1000D fully-connected, softmax			

Figure 2: DenseNet-121 Details

The Dense blocks has varying number of layers with two convolutions. 1*1 sized bottleneck layer and 3*3 sized kernel for convolution. It has 120 convolutions and 4 average pools.

The key advantage is it requires fewer parameters and allows reuse of features resulting in compact models providing better performance compared to Resnet or other models.

4.1.2 CheXNet

CheXNet is model relies on pre-trained weights of ImageNet and trained on NIH chest X-ray images. This model is specifically trained to detect Pneumonia and is based on DenseNet-121

Layers	DenseNet -121	SE-DenseNet-121
Convolution	7*7 conv, stride 2	
Pooling	3*3 max pool, stride 2	
Dense Block (1)	$\begin{pmatrix} 1*1\ conv \\ 3*3\ conv \end{pmatrix} * 6$	$\begin{pmatrix} 1*1\ conv \\ 3*3\ conv \\ fc \end{pmatrix} * 6$
Transition Layer (1)	1*1 conv 2*2 average pool	
Dense Block (2)	$\begin{pmatrix} 1*1\ conv \\ 3*3\ conv \end{pmatrix} * 12$	$\begin{pmatrix} 1*1\ conv \\ 3*3\ conv \\ fc \end{pmatrix} * 12$
Transition Layer (2)	1*1 conv 2*2 average pool	
Dense Block (3)	$\begin{pmatrix} 1*1\ conv \\ 3*3\ conv \end{pmatrix} * 24$	$\begin{pmatrix} 1*1\ conv \\ 3*3\ conv \\ fc \end{pmatrix} * 24$
Transition Layer (3)	1*1 conv 2*2 average pool	
Dense Block (4)	$\begin{pmatrix} 1*1\ conv \\ 3*3\ conv \end{pmatrix} * 16$	$\begin{pmatrix} 1*1\ conv \\ 3*3\ conv \\ fc \end{pmatrix} * 16$
Classification Layer	7*7 global average pool 14D fully-connected, elementwise sigmoid	

Figure 3: CheXNet Details

4.2 Evaluation Metrics

Following metrics will be used for classification,

- **Precision**
  Precision is one indicator of a machine-learning model's performance – the quality of a positive prediction made by the model. Precision refers to the number of true positives divided by the total number of positive predictions (i.e., the number of true positives plus the number of false positives)

- **Recall**
  The recall is calculated as the ratio between the numbers of Positive samples correctly classified as Positive to the total number of Positive samples. The recall measures the model's ability to detect positive samples. The higher the recall, the more positive samples detected. Recall can be thought of as a measure of a classifiers completeness. A low recall indicates many False Negatives.

- **F1-Score**
  The F1 Score is the 2*((precision*recall)/(precision+recall)). It is also called the F Score or the F Measure. Put another way, the F1 score conveys the balance between the precision and the recall.

- **Accuracy**
  Accuracy is a metric for classification models that measures the number of predictions that are correct as a percentage of the total number of predictions that are made. As an example, if 90% of your predictions are correct, your accuracy is simply 90%.

$$Accuracy = \frac{\#\ of\ correct\ predictions}{\#\ of\ total\ predictions}$$

  Accuracy is a useful metric only when you have an equal distribution of classes on your classification. This means that if you have a use case in which you observe more data points of one class than of another, the accuracy is not a useful metric anymore.

- **ROC Curve**
  ROC curve, also known as Receiver Operating Characteristics Curve, is a metric used to measure the performance of a classifier model. The ROC curve depicts the rate of true positives with respect to the rate of false positives, therefore highlighting the sensitivity of the classifier model.

- **IOU (Intersection Over Union)**

  Measures the accuracy of object detection model providing bounding box as output.

  IoU = Area of overlap/Area of Union

  Area of overlap is the overlapped area between predicted bounding box and ground truth bounding box.

  Area of Union is the area comprising both predicted and ground truth bounding box.

- **mAP (Mean Average Precision)**

The mAP value is calculated over recall values of 0 to 1. It compares the ground truth bounding box to predicted set of boxes and returns a score. The score is compared with IOU to decide on the prediction.

4.3 Models Applied

Data generators are used to load the data and pre-process them.

The Models applied:

- DenseNet-121
- DenseNet-121 with CheXNet weights.

4.3.1 DenseNet-121

Summary of model trainable parameters.

Create a `DenseNet121` model

Model: "DenseNet121"

Layer (type)	Output Shape	Param #
DenseNet121 (Functional)	(None, 7, 7, 1024)	7037504
global_average_pooling2d (GlobalAveragePooling2D)	(None, 1024)	0
dropout (Dropout)	(None, 1024)	0
dense (Dense)	(None, 1)	1025

Total params: 7,038,529
Trainable params: 6,954,881
Non-trainable params: 83,648

There are total of 6,954,881 trainable parameters.

The model was executed for 10 epochs with batch size of 32. The binary cross entropy loss is around 0.3 and validation accuracy of 85%.

The results for model run on evaluation data are,

Evaluate the model on validation data

Loss: 0.338, Accuracy: 0.843, Average Precision: 0.665, F1 Score: 0.612

AUC: 0.890

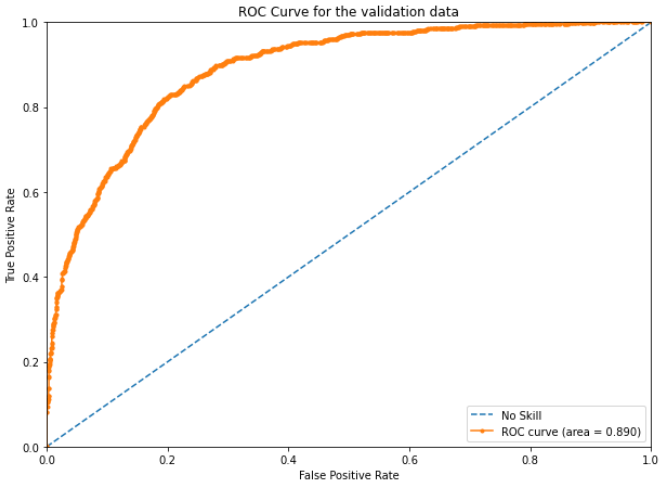


Figure 4: ROC for Validation data

ROC Curve for the test data

AUC: 0.894

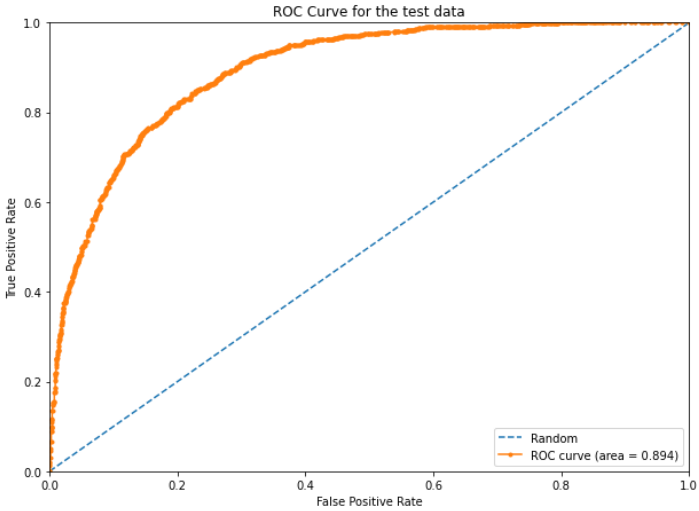


Figure 5:ROC for test data

The AUC is similar for validation and test data.

The confusion matrix for test data indicating the predictions is shown below.

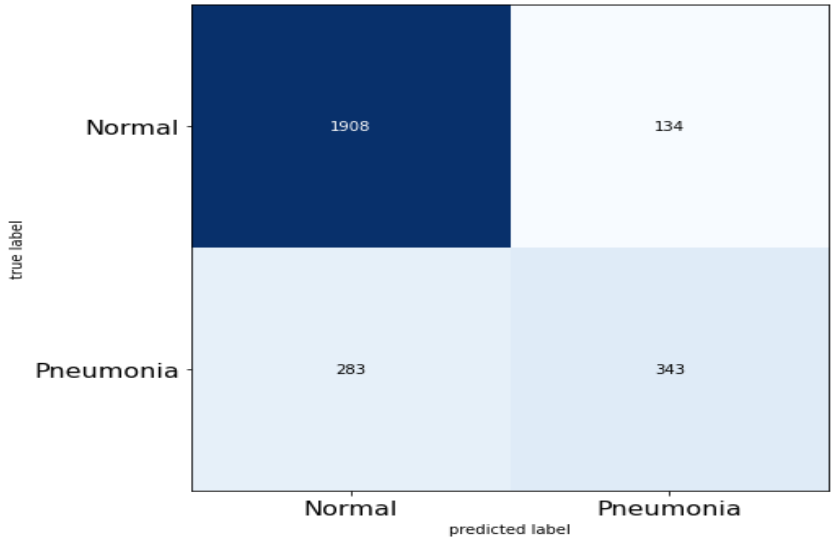


Figure 6: CM for test data

The classification report for test data :

Classification Report on the test data

---

	precision	recall	f1-score	support
Normal	0.87	0.93	0.90	2042
Pneumonia	0.72	0.55	0.62	626
accuracy			0.84	2668
macro avg	0.79	0.74	0.76	2668
weighted avg	0.84	0.84	0.84	2668

The F1-score for Normal class is 0.9 indicating a high accuracy. The F1-score of 0.62 is low for Pneumonia class which is the main objective of the model.

4.3.2 DenseSet-121 with CheXNet weights

The summary of model:

Create a `CheXNet-like` model using pre-trained weights

Model: "CheXNet-like"

Layer (type)	Output Shape	Param #
CheXNet-like (Model)	(None, 1024)	7037504
dropout (Dropout)	(None, 1024)	0
dense (Dense)	(None, 1)	1025

==========
Total params: 7,038,529
Trainable params: 6,954,881
Non-trainable params: 83,648
_____

The loss is around ~0.32 and accuracy is 85% after model execution on training samples for 10 epochs with batch size of 32.

Summary for evaluation on validation data:

Evaluate the model on validation data
----------Loss: 0.325, Accuracy: 0.848, Average Precision: 0.692, F1 Score: 0.61

ROC Curve for validation data.

AUC: 0.898

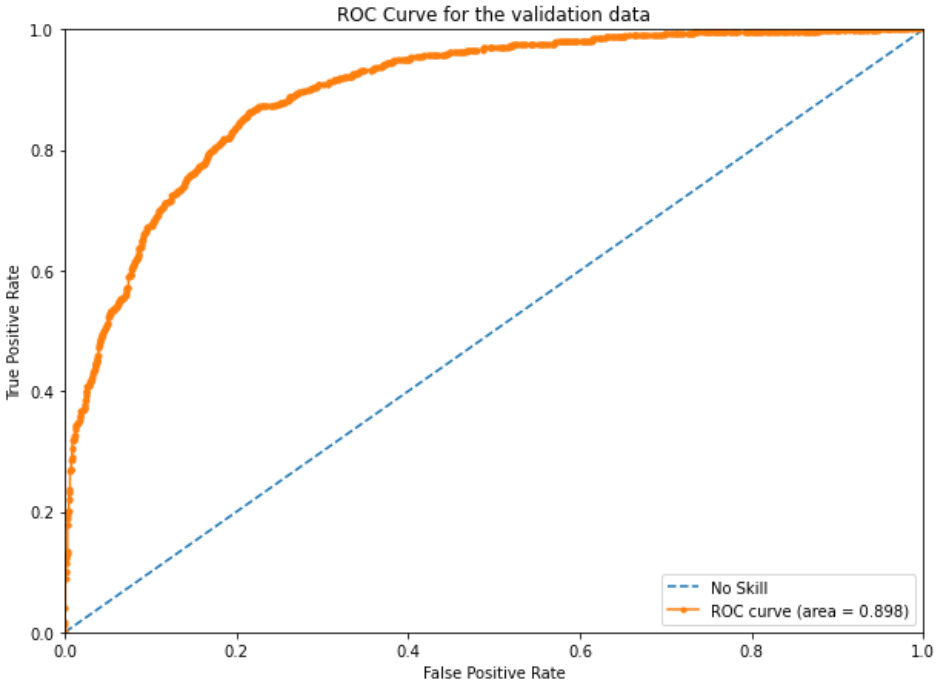


Figure 7:ROC for validation data

ROC curve for test data:

AUC:0.895

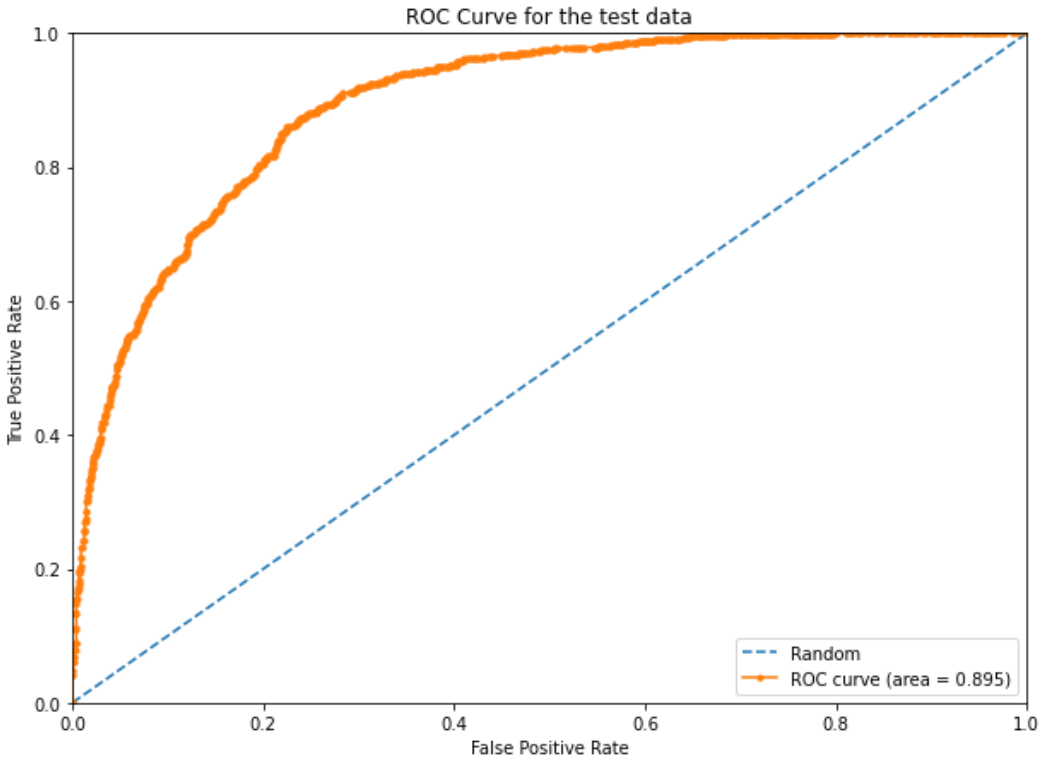


Figure 8: ROC for test data

The confusion matrix for predicted labels:

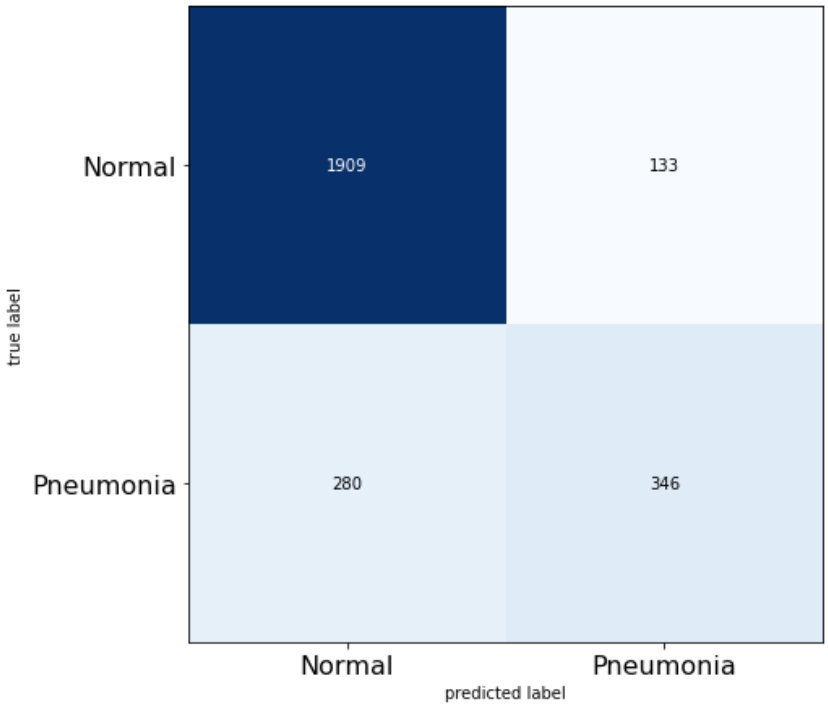


Figure 9: CM for predicted class

The classification report on test data:

Classification Report on the test data

---

	precision	recall	f1-score	support
Normal	0.87	0.93	0.90	2042
Pneumonia	0.72	0.55	0.63	626
accuracy			0.85	2668
macro avg	0.80	0.74	0.76	2668
weighted avg	0.84	0.85	0.84	2668

The classification report shows that F1-score for Pneumonia class is still low showing a very marginal improvement.

4.4 Loss/Accuracy Graphs

The graphs show the variation in loss and accuracy during training and validation for denseNet-121 and densenet-121 with Chexnet weights.

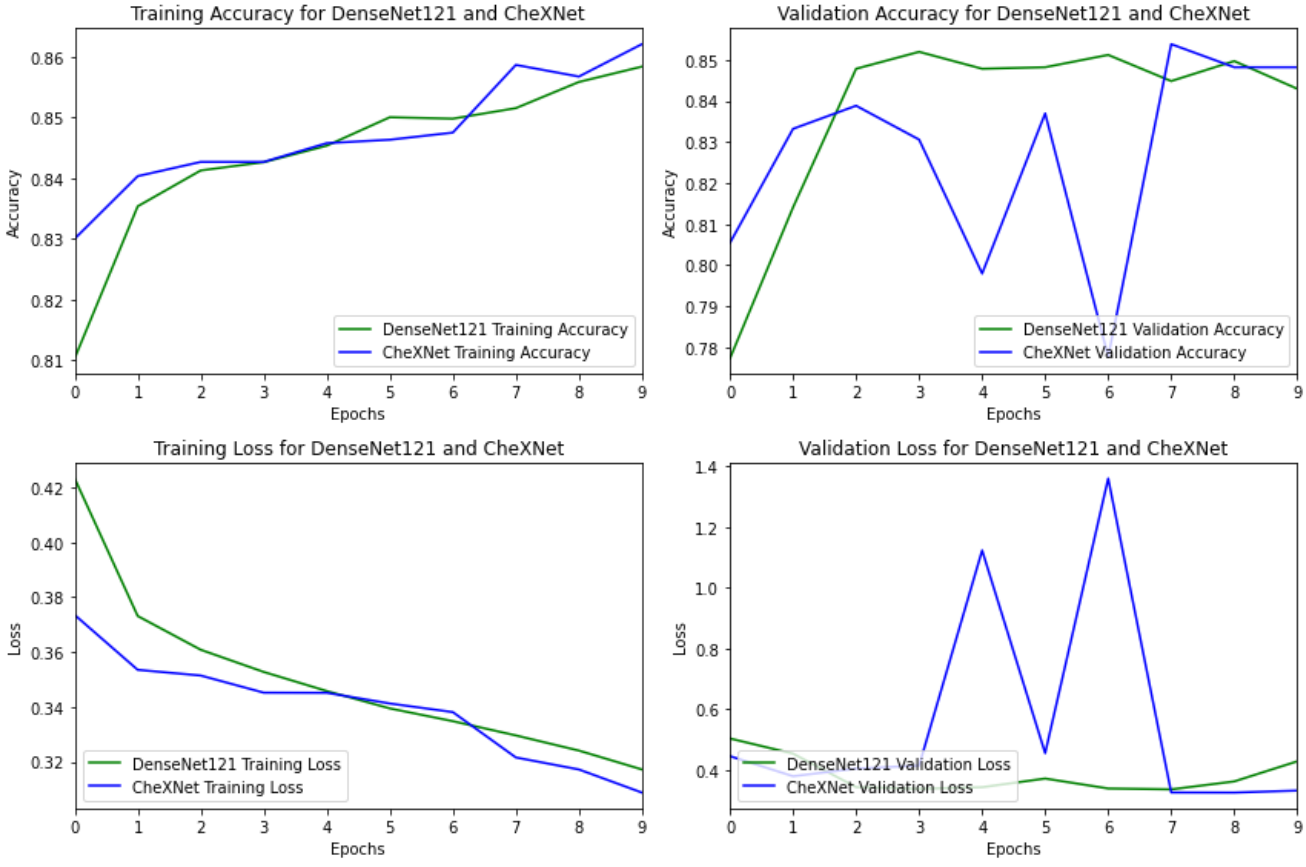


Figure 10:DenseNet vs Chexnet loss & accuracy

Observations:

- It is observed that training accuracy for DenseNet-121 is starting a lower value than the model with CheXnet weights. At the end of 10 epochs, both models reach similar accuracy.
- The validation accuracy swings low and high during epochs where as DenseNet-121 accuracy goes high and then stays around a median.
- Training loss for DenseNet121 is high initially but in the end goes down to 0.33.Loss is slightly better when CheXNet weights are used.
- Swings are observed when ChexNet weights are used in validation loss. In the end, validation loss is better when Chexnet weights are used.

4.5 How to improve performance

Since the F1-score for Pneumonia class is around 0.63 and accuracy around 85% is not optimal for the given problem, following methods can be applied to improve performance,

- Data augmentation.
  - Rotate.
  - Flip.
  - Masking.
- FCNN model.
- CNN model.
- Apply transfer-learning methods, using MobileNetV2 model

4.5.1 FCNN Model

Summary of model trainable parameters:

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 flatten (Flatten)           (None, 49152)             0

 dense (Dense)               (None, 128)               6291584

 dense_1 (Dense)             (None, 1)                 129

=================================================================
Total params: 6,291,713
Trainable params: 6,291,713
Non-trainable params: 0
_________________________________________________________________
```

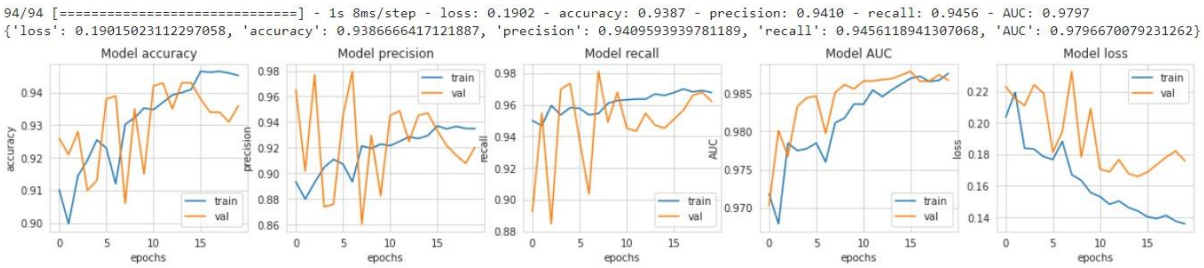


Figure 11: Graphs for Fast CNN

4.5.3 CNN Model

In Convolutional Neural Network, the model first learns the features from the images and based on that it helps in classification.

The reason for popularity of CNN is its very good execution while classifying. CNN was founded on the idea of recognizing handwritten zip codes. It is a feed-forward kind of NN. Composition of multiple layers of artificial neurons forms Convolutional neural networks.

There is first the Convolutional Layer: which is responsible for learning the characteristics of the images input,

Then there is Max-Pooling Layer: it performs max-pooling operation by selecting max value from kernel matrix,

Then comes Fully Connected Layer: all the neurons of a particular layer are connected to the neurons of the next layer to prepare the network for classification capabilities.

Summary of model trainable parameters:

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 conv2d (Conv2D)              (None, 126, 126, 32)      896

 max_pooling2d (MaxPooling2D  (None, 63, 63, 32)        0
 )

 conv2d_1 (Conv2D)            (None, 61, 61, 32)        9248

 max_pooling2d_1 (MaxPooling  (None, 30, 30, 32)        0
 2D)

 dropout (Dropout)            (None, 30, 30, 32)        0

 conv2d_2 (Conv2D)            (None, 28, 28, 64)        18496

 conv2d_3 (Conv2D)            (None, 26, 26, 128)       73856

 batch_normalization (BatchN  (None, 26, 26, 128)       512
 normalization)
```

```
max_pooling2d_2 (MaxPooling2D)   (None, 13, 13, 128)      0

dropout_1 (Dropout)              (None, 13, 13, 128)      0

flatten_1 (Flatten)              (None, 21632)            0

dense_2 (Dense)                  (None, 512)              11076096

batch_normalization_1 (Batch     (None, 512)              2048
Normalization)

dropout_2 (Dropout)              (None, 512)              0

dense_3 (Dense)                  (None, 128)              65664

batch_normalization_2 (Batch     (None, 128)              512
Normalization)

dropout_3 (Dropout)              (None, 128)              0

dense_4 (Dense)                  (None, 1)                129

=================================================================
Total params: 11,247,457
Trainable params: 11,245,921
Non-trainable params: 1,536
_________________________________________________________________
```

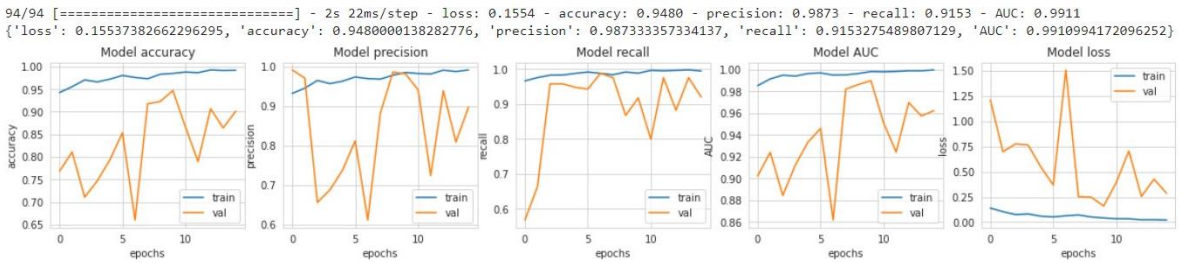


Figure 12: Graphs for CNN

4.5.4 MobileNetV2 Model with Transfer Learning

Transfer learning is defined as the use of a pre-trained model trained already on a huge/certain dataset for training newer datasets. The weights of pre-trained model are reused avoiding re-training for the new dataset.

Using a pre-trained model as the starting point for some particular and related tasks , in another expression, helps to save time and computing resources. Since the CNN learned to extract features from images in the initial training process and depending on the capability to extract the most significant and important ones. In the next phase and during the new training the CNN and according to its past knowledge in features extraction, which was

obtained during the original training, there are two ways to utilize the capabilities of the pre-trained CNN.

- First way is to use the pre-trained convolutional neural network as a feature extractor, this technique is called feature extraction via transfer learning, what differentiates it is that the classification operation uses weighs and features from the precedent extraction and feed it into a new network that performs the classification task.

- Second way (used in this paper) and a more sophisticated procedure is to retain specific knowledge mined from the previous task and to feed it into a modified CNN architecture with the tuning of the trainable parameters.

Summary of model trainable parameters:

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-appli
cations/mobilenet_v2/mobilenet_v2_weights_tf_dim_ordering_tf_kernels_1.0_12
8_no_top.h5
9412608/9406464 [==============================] - 0s 0us/step
9420800/9406464 [==============================] - 0s 0us/step
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 mobilenetv2_1.00_128 (Funct  (None, 4, 4, 1280)       2257984
 ional)

 global_average_pooling2d (G  (None, 1280)             0
 lobalAveragePooling2D)

 dense_5 (Dense)             (None, 1)                 1281

=================================================================
Total params: 2,259,265
Trainable params: 1,281
Non-trainable params: 2,257,984
_________________________________________________________________
```

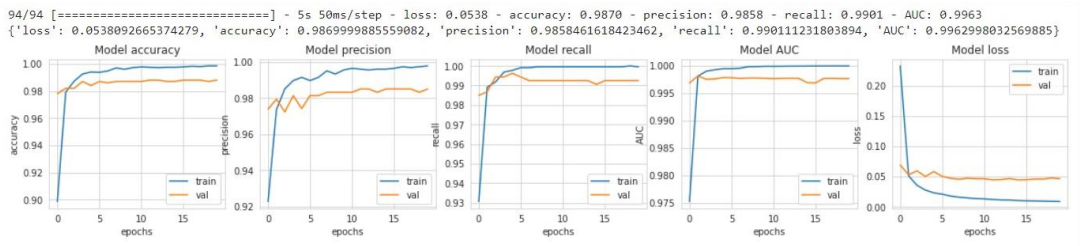


Figure 13: Graphs for MobileNetV2

4.5.5 Image Augmentation with Mobile NET

- Image augmentation is a technique of altering the existing data to create some more data for the model training process. In other words, it is the process of artificially expanding the available dataset for training a deep learning model
- We performed data augmentation by transforming the pictures (random rotation, flipping, random grayscale, and horizontal and vertical shifting) to produce more diversity in the dataset.
- Similar as above, we tried training the dataset through Mobile Net transfer learning technique. The dataset was split into train, validation, and test.
- Few samples image after augmentation as shown below.

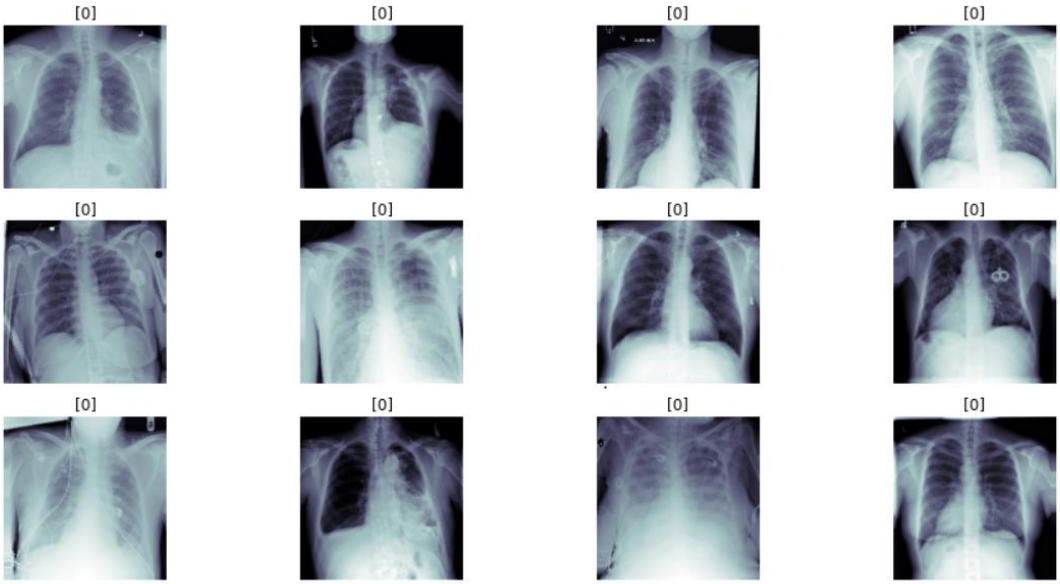


Figure 14: Augmented Images

- The model was trained with the base imagenet dataset followed by a dense layer of MobileNet.
- Callbacks used for this case was CSVlogger, ReduceLROnPlateau.
- Below screenshots talks about the model overview, AUC curve and confusion matrix readings.
- The results were not satisfactory with image augmentation as the model was highly overfitting.

Summary of model trainable parameters:

```
    model.save(MODEL_WEIGHTS + FINAL_MODEL)
```

```
Lets fit the model.....
Create a `MobileNet` model
--------------------------------------------------------------------------------
WARNING:tensorflow:`input_shape` is undefined or non-square, or `rows` is
d as the default.
Model: "MobileNet"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 MobileNet (Functional)      (None, 7, 7, 1024)        3228864

 global_average_pooling2d_1   (None, 1024)             0
 (GlobalAveragePooling2D)

 dropout_1 (Dropout)         (None, 1024)              0

 dense_1 (Dense)             (None, 1)                 1025

=================================================================
Total params: 3,229,889
Trainable params: 3,208,001
Non-trainable params: 21,888
```

```
ROC Curve for the test data
--------------------------------------------------------------------------------
AUC: 0.490
```

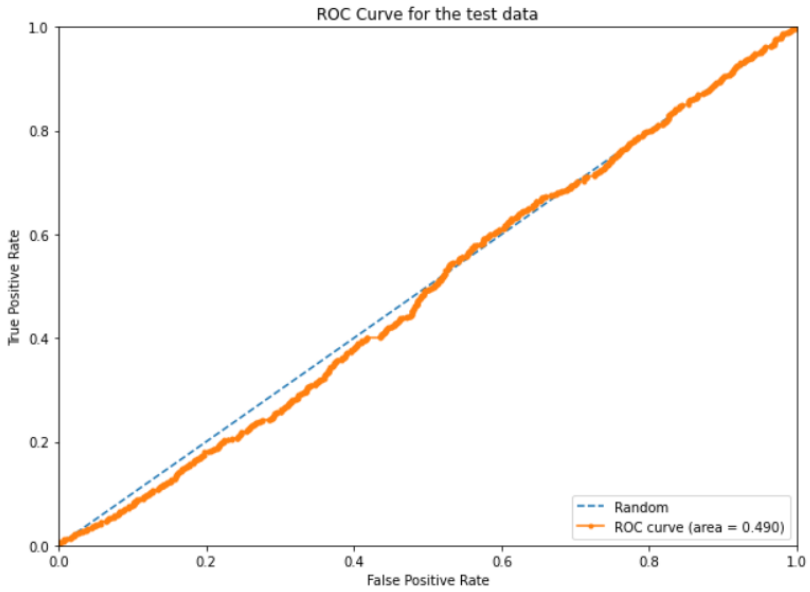
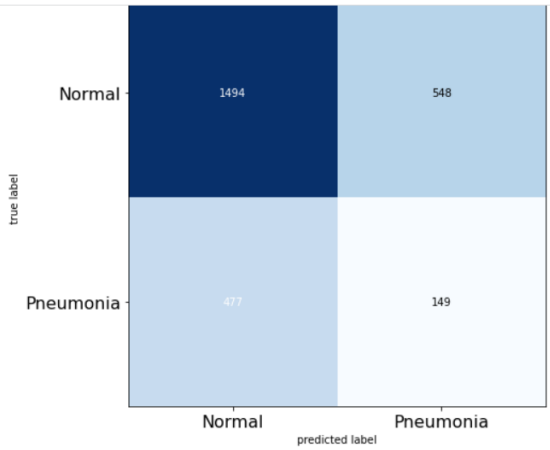


Figure 15: ROC Curve for augmented data




```
[40]: from sklearn.metrics import recall_score, confusion_matrix, precision_score, f1_score, accuracy_score, roc_
      from sklearn.metrics import classification_report

      print('Classification Report on the test data'); print('--'*60)
      print(classification_report(y_test, test_pred, target_names = ['Normal', 'Pneumonia']))
```

```
Classification Report on the test data
--------------------------------------------------------------------------------
              precision    recall  f1-score   support

      Normal       0.76      0.73      0.74      2042
   Pneumonia       0.21      0.24      0.23       626

    accuracy                           0.62      2668
   macro avg       0.49      0.48      0.48      2668
weighted avg       0.63      0.62      0.62      2668
```

Figure 16: CM for augmented data

4.5.6 MobileNetV2 with classification and localization

MobileNetV2 is described in section 4.5.4. Since the model of section 4.5.4 focused on classification, two additional layers are added to determine the bounding box as well i.e a flatten layer and dense layer with 4 neurons. The model is shown below.

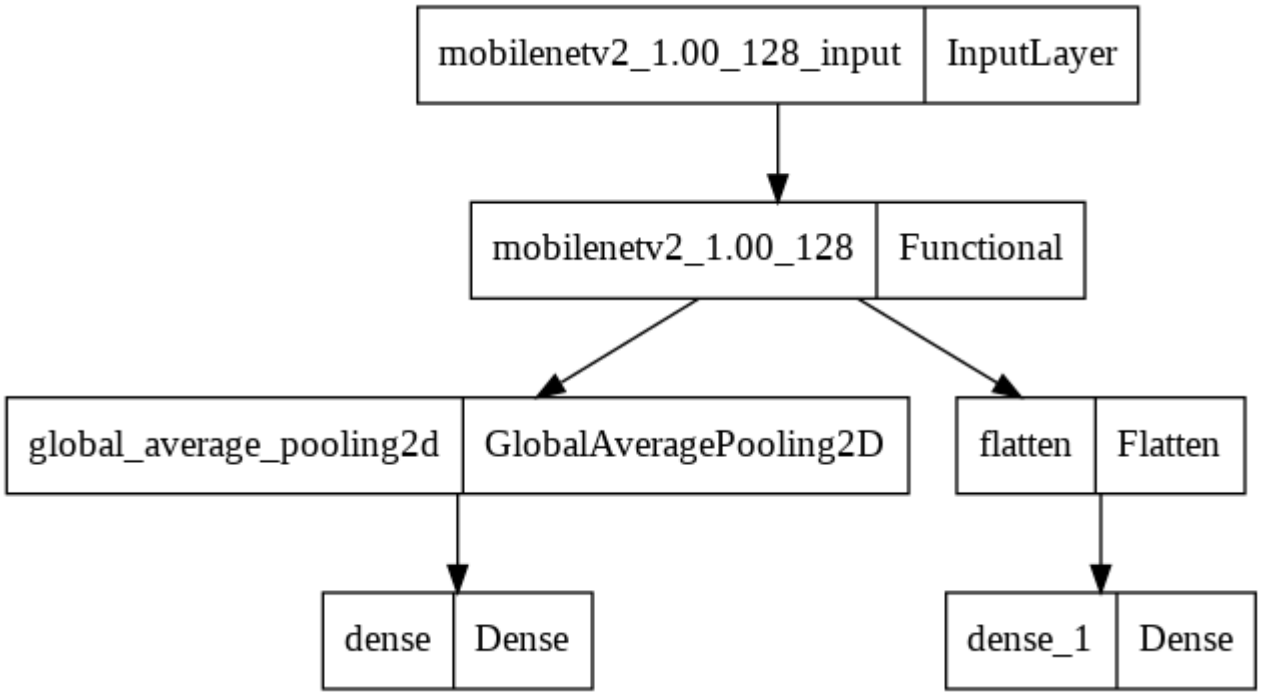


Figure 17: MobileNetV2 with additional localization layer

```
Total params: 2,341,189
Trainable params: 83,205
Non-trainable params: 2,257,984
```

The model stops learning after 5-10 epochs. The below graphs indicate the same.

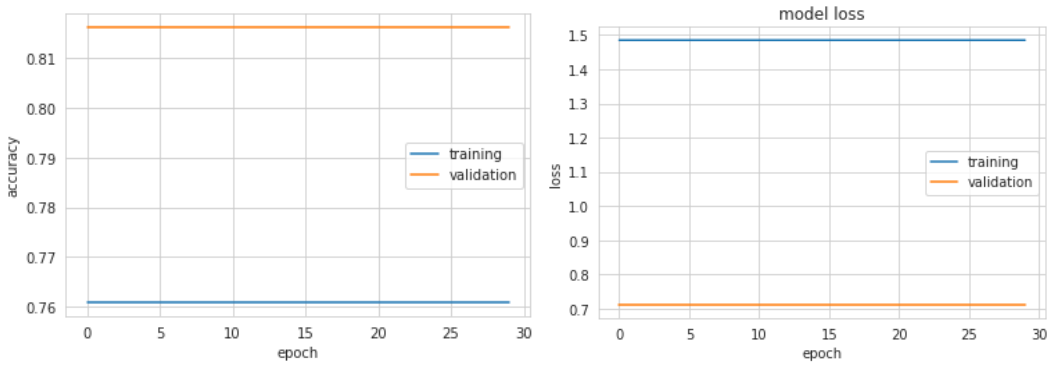


Figure 18: Accuracy and Loss graph

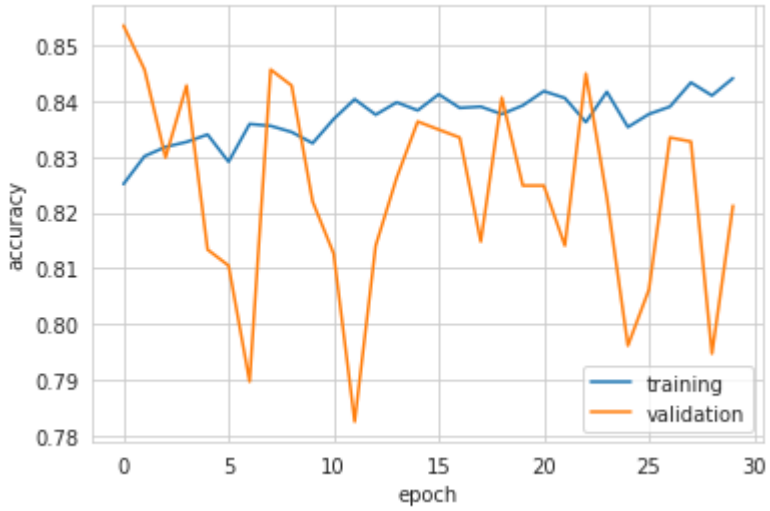


Figure 19: Accuracy history over epocs

Figure 20: Loss over epochs

The graph for classification accuracy indicates that it remains steady for training data but varies significantly for validation data. The accuracy for validation data oscillates a lot. The loss also oscillates a lot for validation while remaining linear for training data. This also implies model has stopped learning after few epochs. In case of bounding box detection, validation accuracy is better than training accuracy. The training loss is more than validation loss.

This model is not able to predict the bounding box correctly.

4.5.7 Unet with Mobilenet

MobileNet in combination with Unet is used to detect and localize the area of lung opacity and hence Pneumonia.

The model summary is as below.

```
Total params: 3,230,849
Trainable params: 3,208,961
Non-trainable params: 21,888
```

In this approach data generators are used to read and pre-process the data in batches to minimize memory consumption. IN order to reduce the time to run the model, 6000 images are used for training and 2000 images are used for validation.

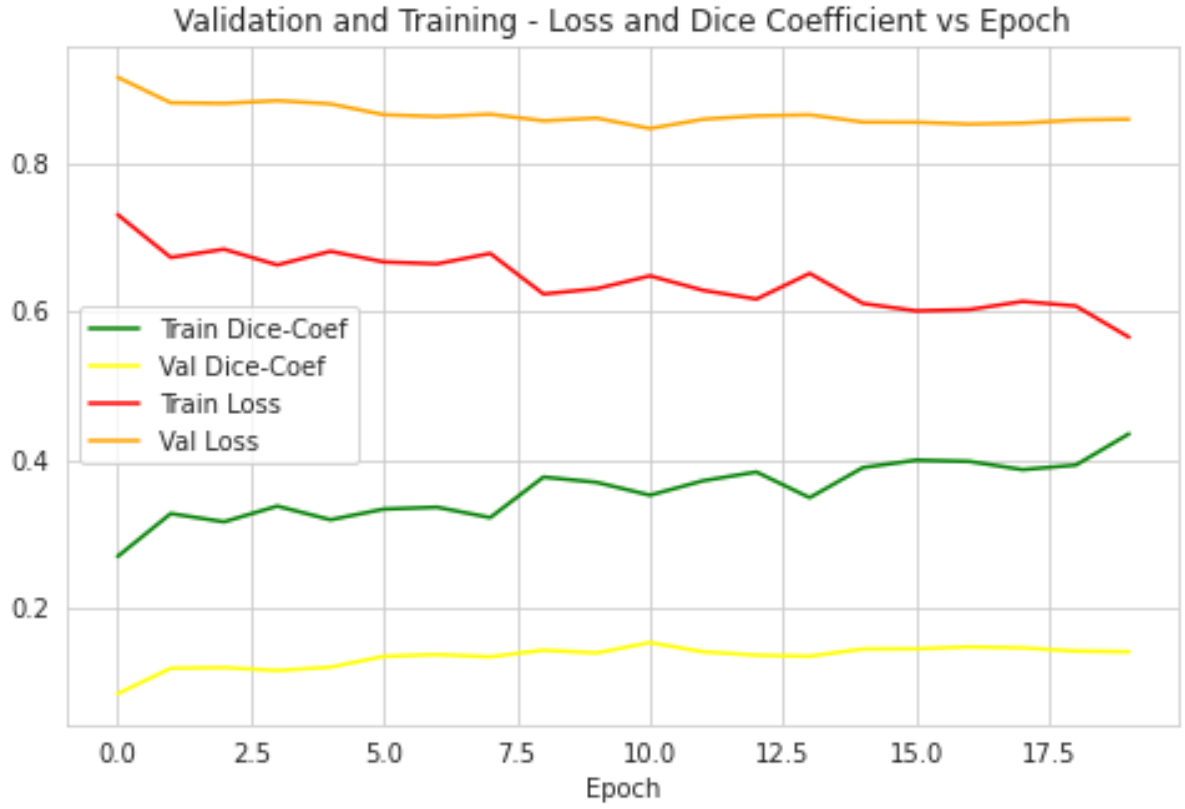


Figure 21: Graph for Unet with Mobilenet

Some of the predicted results are shown below.

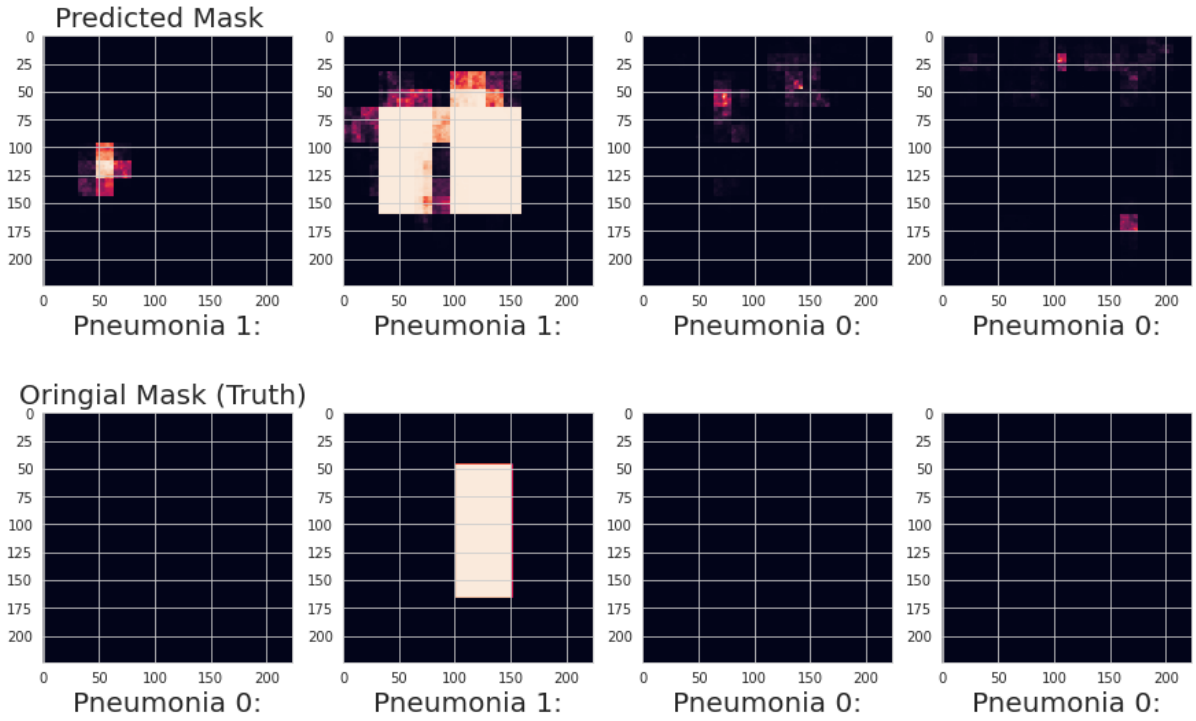


Figure 22: Predictions on sample test data

The confusion matrix for 20 images is shown below.

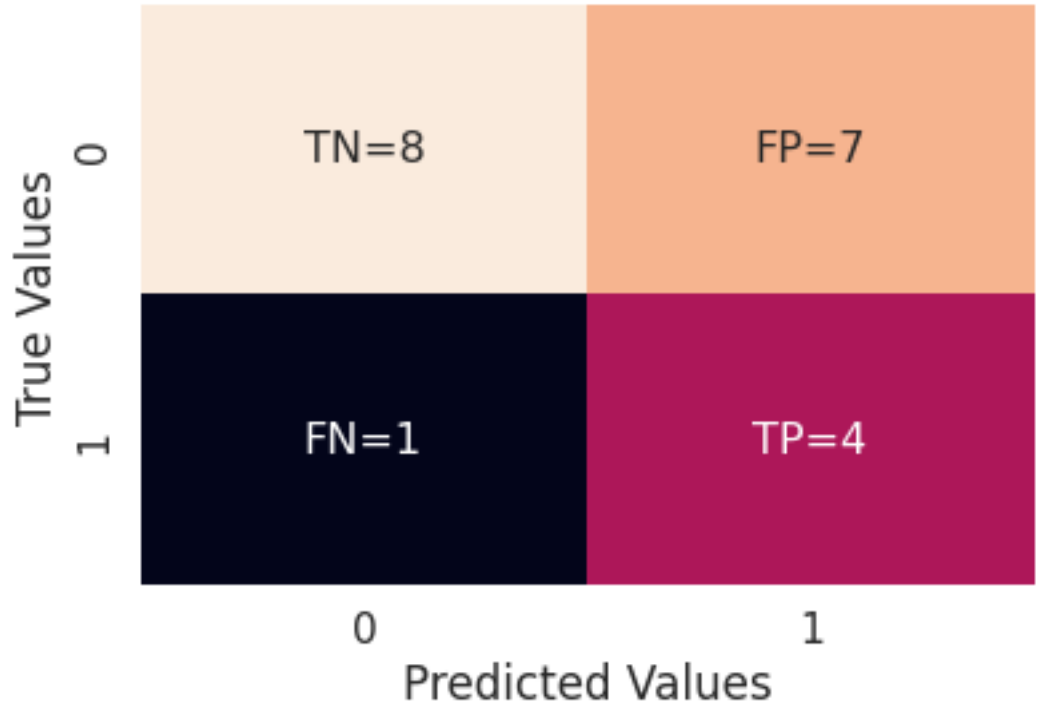


Figure 23: CM for Unet with Mobilenet

```
              precision    recall  f1-score   support

           0       0.89      0.53      0.67        15
           1       0.36      0.80      0.50         5

    accuracy                           0.60        20
   macro avg       0.63      0.67      0.58        20
weighted avg       0.76      0.60      0.63        20
```

The unet model with mobilenet is able to detect and predict the bounding box with 50% accuracy for Pneumonia class. The accuracy can further be increased by running model without resizing the image and for longer epochs.

4.6 Results Summary

S. No.	Model	Class	Accuracy	Recall	Precision	F1-Score
1	DenseNet 121	Normal	84%	0.93	0.87	0.9
2	DenseNet 121	Pneumonia	84%	0.55	0.72	0.62
3	DenseNet121+CheXnet weights	Normal	85%	0.93	0.87	0.9
4	DensetNet121+CheXnet weights	Pneumonia	85%	0.55	0.72	0.63
5	FCNN	Pneumonia	93.9%	0.95	0.94	0.95
6	CNN	Pneumonia	94.8%	0.91	0.98	0.94
7	MobileNetV2 with Transfer learning	Pneumonia	98.7%	0.98	0.99	0.98
8	MobileNet with image augmentation	Pneumonia	62%	0.24	0.21	0.23
9	MobileNetv2 with transfer learning and 1 dense layer for localization		82%			
10	MobileNet with unet	Pneumonia	50%	0.36	0.80	0.50

5. Implications

5.1 What causes Pneumonia?

There are more than 30 different causes of pneumonia, and they are grouped by the cause. The main types of pneumonia are:

- **Bacterial pneumonia.** This type is caused by various bacteria. The most common is *Streptococcus pneumoniae*. It usually occurs when the body is weakened in some way, such as by illness, poor nutrition, old age, or impaired immunity, and the bacteria can work their way into the lungs. Bacterial pneumonia can affect all ages, but you are at greater risk if you abuse alcohol, smoke cigarettes, are debilitated, have recently had surgery, had a respiratory disease or viral infection, or had a weakened immune system.
- **Viral pneumonia.** This type is caused by various viruses, including the flu (influenza), and is responsible for about one-third of all pneumonia cases. You may be more likely to get bacterial pneumonia if you have viral pneumonia.
- **Mycoplasma pneumonia.** This type has somewhat different symptoms and physical signs and is referred to as atypical pneumonia. It is caused by the bacterium *Mycoplasma pneumoniae*. It generally causes a mild, widespread pneumonia that affects all age groups.
- **Other pneumonias.** There are other less common pneumonias that may be caused by other infections including fungi.

5.2 Who is at risk for pneumonia?

Anyone can get pneumonia. However, the following groups are at the highest risk:

- Adults aged 65 and older
- Children younger than age 2
- People with certain medical conditions
- People that smoke

5.3 How is Pneumonia treated?

Treatment depends on the type of pneumonia you have. Most of the time, pneumonia is treated at home, but severe cases may be treated in the hospital. Antibiotics are used for bacterial pneumonia. Antibiotics may also speed recovery from mycoplasma pneumonia and some special cases. Most viral pneumonias don't have specific treatment. They usually get better on their own.

Other treatment may include eating well, increasing fluid intake, getting rest, oxygen therapy, pain medicine, fever control, and maybe cough-relief medicine if cough is severe.

5.4 Complications of pneumonia?

Most people with pneumonia respond well to treatment, but pneumonia can be very serious and even deadly.

You are more likely to have complications if you are an older adult, a very young child, have a weakened immune system, or have a serious medical problem like diabetes or cirrhosis. Complications may include:

- **Acute respiratory distress syndrome (ARDS).** This is a severe form of respiratory failure.
- **Lung abscesses.** These are pockets of pus that form inside or around the lung. They may need to be drained with surgery
- **Respiratory failure.** This requires the use of a breathing machine or ventilator.
- **Sepsis.** This is when the infection gets into the blood. It may lead to organ failure.

5.5 Key implications about pneumonia

- Pneumonia is an infection of one or both lungs caused by bacteria, viruses, or fungi.
- There are more than 30 different causes of pneumonia, and they're grouped by the cause. The main types of pneumonia are bacterial, viral, and mycoplasma pneumonia.
- A cough that produces green, yellow, or bloody mucus is the most common symptom of pneumonia. Other symptoms include fever, shaking chills, shortness of breath, low energy, and extreme tiredness.
- Pneumonia can often be diagnosed with a thorough history and physical exam. Tests used to look at the lungs, blood tests, and tests done on the sputum you cough up may also be used.
- Treatment depends on the type of pneumonia you have. Antibiotics are used for bacterial pneumonia. It may also speed recovery from mycoplasma pneumonia and some special cases.

  Most viral pneumonias don't have a specific treatment and just get better on their own. Other treatment may include a healthy diet, more fluids, rest, oxygen therapy, and medicine for pain, cough, and fever control.

- Most people with pneumonia respond well to treatment, **but pneumonia can cause serious lung and infection problems. It can even be deadly.**

6. Limitations

Our model seems to have high recall and AUC score, but there could still be blind spots due to the limitations of dataset. Given more time and resources, we can explore followings:

1. **Include more diversity in the dataset in terms of patient residence and age**
   our current model is trained with dataset that only consists of X-Ray images of patients who are from a particular geographical location. This brings up the question on whether our model can accurately detect Pneumonia if patients do not reside in a particular geographical location. Make the model more robust, we must include more diversity in dataset, i.e., including X-Ray images from patients in various parts of world.

2. Train the model with dataset that is more representative to patient population distribution in terms of condition complexity
   So far, we built a binary classification model, identifying whether the patient has Pneumonia or not. Patients could be suffering from diverse types of lung

diseases (sometimes more than one at a time) and they would still show up as an infiltrate on the X-Rays. We must combine our current dataset with dataset from NIH Clinical Centre, which consists over 110,000 chest X-ray images of more than 30,000 patients, with 14 common thoracic disease labels. The NIH team believes that the dataset would be "significantly more representative to the real patient population distributions and realistic clinical diagnosis challenges."

3. Separation of COVID-19 from non-COVID pneumonia is a more challenging task than the separation of pneumonia from other lung pathologies. The expansion of the pneumonia class also raises concerns as to whether COVID-19 cases are truly being distinguished from pneumonia cases as reported, or if they are being separated from the alternative classes included in the pneumonia class of COVID-19.

7. Closing Reflections

In this paper, five mainstream deep learning models are used to diagnose clinical data on a dataset consisting of X-ray images of the lungs with pneumonia and normal lungs and the accuracy of these methods are compared. Among them, because of the superior performance of MobileNetV2, we focus on the network structure of MobileNetV2. The results demonstrated that all five-network structures can recognize pneumonia and the accuracy of MobileNetV2 is higher than other network structures. In addition, the application of artificial intelligence technology in the medical field is not sufficient, and the dataset in this field should be improved in terms of types. As the amount of pneumonia image data increases and the network structure continues to improve, the performance of CNN-based pneumonia diagnosis algorithms will also continue to improve. In the future, the application of clinical image diagnosis of pneumonia X-rays can reduce the workload of clinicians and enable patients to obtain early diagnosis and timely treatment, thereby reducing the mortality rate of pneumonia.

From the experimental results, MobileNetV2, as a lightweight network, not only has a smaller number of calculations than most CNNs but also has a better classification effect than other types of CNN models when the number of parameters is almost on an order of magnitude. This benefits from using the depth wise separable convolution. Since the development of deep learning, most image recognition models have large parameters and a large number of calculations, which are not suitable for use in embedded devices. For the identification of pneumonia, a common disease, we must also consider how to identify pneumonia quickly and accurately in areas where equipment and doctors are scarce. This is one of the reasons why we recommend using MobileNetV2 for pneumonia recognition.