

CAPSTONE PROJECT REPORT

BY

Vishnupriya Vijayan

Parag Vivarekar

Padmavathy Kandhan

Jay Patadia

Lola babu Agam

Krishnani

INDEX

| S. No | CONTENT | Pg. No |
|-------|---|--------|
| 1 | Summary of problem statement, data and findings | 2 |
| 2 | Overview of the final process | 5 |
| 3 | Step-by-step walk through the solution | 13 |
| 4 | Model evaluation | 18 |
| 5 | Comparison to benchmark | 25 |
| 6 | Implications | 25 |
| 7 | Limitations | 26 |
| 8 | Closing Reflections | 26 |

SUMMARY OF PROBLEM STATEMENT, DATA AND FINDINGS

In medicine, the next frontier for AI is anomaly localization in medical imaging. Localization of anomalies refers to both predicting anomalies and their boundaries. Automatic detection algorithms to locate inflammation in an image can help physicians make better clinical decisions. In this project, we analyze data with the knowledge of EDA. We build a detection model and present our findings based on the evaluations with the RSNA Pneumonia Detection Challenge dataset.

OVERVIEW OF PNEUMONIA

Pneumonia is a form of an acute respiratory infection that affects the lungs. The lungs comprise small sacs called alveoli that fill up with oxygen as a healthy person breathes. The alveoli are filled with pus and fluid when a person has pneumonia, making breathing difficult, and reducing oxygen intake.



Figure 1- X-Ray Image of Lungs with Pneumonia (sample image from Dataset)

The single most significant bacterial cause of death in children worldwide is pneumonia. In 2017, pneumonia killed 808,694 children under the age of 5, accounting for 15 percent of all deaths by children under five. Children and families worldwide are afflicted by pneumonia, but it is most

common in South Asia and sub-Saharan Africa. It can be avoided with easy procedures and managed with low-cost, low-tech treatment and care.

In 2015 spending for maternal, infant, and child survival, the cost of antibiotic care for all children with pneumonia estimate at about US\$ 109 million per year among 66 countries. The expense requires antibiotics and diagnostics for the treatment of pneumonia.

CAUSES AND TRANSMISSION

According to WHO, pneumonia is caused by several infectious agents, including viruses, bacteria, and fungi. The most common are:

- Streptococcus pneumoniae* – the most common cause of bacterial pneumonia in children;
- Haemophilus influenzae* type b (Hib) – the second most common cause of bacterial pneumonia;
- the respiratory syncytial virus is the most common viral cause of pneumonia;
- in infants infected with HIV, *Pneumocystis jiroveci* is one of the most common reasons for pneumonia, responsible for at least one-quarter of all pneumonia deaths in HIV- infected infants.

Spreading of Pneumonia happens in many ways. The viruses and bacteria commonly found in a child's nose or throat can infect the lungs while breathing. They may also spread via air- borne droplets from a cough or sneeze. Besides, pneumonia may spread through blood, especially during and shortly after birth. More research needs to be done on the different pathogens causing pneumonia and how they are transmitted, as this is of critical importance for treatment and prevention.

TREATMENT AND PREVENTION

Pneumonia is treated with antibiotics. Amoxicillin-dispersible tablets are the antibiotic of choice. In most pneumonia cases, oral antibiotics are needed, which are mostly administered at a health clinic. These cases may also be diagnosed and treated at the neighborhood level by qualified community health professionals with affordable oral antibiotics. Only for severe cases of pneumonia is hospitalization recommended.

DIAGNOSTIC PROCEDURE

The doctor will diagnose pneumonia based on your medical history, a physical exam, and test results. Sometimes pneumonia is hard to analyze because symptoms may be the same as a cold or flu. The patient may not realize that his/her condition is more severe until it lasts longer than these other conditions.

If the doctor thinks the patient may have pneumonia, they may do one or more of the following tests.

- Chest X-ray to look for inflammation in the patient's lungs. A chest X-ray is often used to diagnose pneumonia.
- Blood tests, such as a complete blood count (CBC), determine whether the patient's immune system is fighting an infection.
- Pulse oximetry to measure how much oxygen is in his/her blood. Pneumonia can keep the patient's lungs from moving enough oxygen into his/her blood. A small sensor called a pulse oximeter is attached to the patient's finger or ear to calculate the levels.

DATA DESCRIPTION

In 2018, RSNA organized an AI challenge to detect pneumonia, one of the leading causes of mortality worldwide, as part of its efforts to help improve artificial intelligence (AI) instruments for radiology. RSNA Pneumonia dataset consists of 29684 thousand images. All the images are in Dicom format. There are 3000 images for testing and the remaining for training.

Dicom images: The images are in a particular format called DICOM files (*.dcm). They contain a mix of header metadata as well as pixel data underlying raw image arrays.

There are three classes in the dataset - Normal, Not normal/No opacity, and Lung opacity. Normal class indicates there is no anomaly in the lungs. Not normal/No opacity demonstrates to those who do not have pneumonia, but the image still has some abnormality. Sometimes, this finding could mimic the appearance of the right pneumonia. Lung opacity class indicates there is definite pneumonia in the lungs. Finally, these three classes are divided into two target variables, 0 and 1. The images with lung opacity come under target 1 and 0 for the other two classes.

Along with the images, two csv files are provided. A detailed class info file consists of the image name and the class it belongs to. The train labels file consists of the bounding box coordinates belonging to each image. Bounding box coordinates are given in the following format:

- x -- the upper-left x coordinate of the bounding box.
- y -- the upper-left y coordinate of the bounding box.
- width -- the width of the bounding box.
- height -- the height of the bounding box.

With these bounding box coordinates, the target column is provided, which discriminates classes into categories of 0 and 1.

OVERVIEW OF THE FINAL PROCESS

EDA AND PREPROCESSING

Train Labels Dataset

There are 20672 instances with a target value of 0 (indicating no pneumonia) and 9555 instances with a target value of 1 (indicating pneumonia presence).

```
1 #lets begin with train csv convert to dataframe
2 train_df = pd.read_csv('stage_2_train_labels.csv')
3 train_df
```

| | patientId | x | y | width | height | Target |
|---|--------------------------------------|-------|-------|-------|--------|--------|
| 0 | 0004cfab-14fd-4e49-80ba-63a80b6bddd6 | NaN | NaN | NaN | NaN | 0 |
| 1 | 00313ee0-9eaa-42f4-b0ab-c148ed3241cd | NaN | NaN | NaN | NaN | 0 |
| 2 | 00322d4d-1c29-4943-afc9-b6754be640eb | NaN | NaN | NaN | NaN | 0 |
| 3 | 003d8fa0-6bf1-40ed-b54c-ac657f8495c5 | NaN | NaN | NaN | NaN | 0 |
| 4 | 00436515-870c-4b36-a041-de91049b9ab4 | 264.0 | 152.0 | 213.0 | 379.0 | 1 |

Figure 1- Shape of the data

From Figure 2 we can see that stage_2_train_labels.csv file contains patientId, which is a unique value per patient. Each patientId has one target column and four values: the corresponding abnormality bounding box defined by the upper-left-hand corner ‘x’ and ‘y’ coordinate and its corresponding width and height. The target column has two values 0 and 1.

Class Info Dataset

```
1 #now checking with second detail class csv and dump into dataframe
2 df_class = pd.read_csv('stage_2_detailed_class_info.csv')
3 df_class.info()
4

>
RangeIndex: 30227 entries, 0 to 30226
Data columns (total 2 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   patientId  30227 non-null   object 
 1   class       30227 non-null   object 
dtypes: object(2)
```

Figure 2- Class Info Dataset

Consists of 3 classes mainly for each patientId. 0 is for No Lung Opacity / Not Normal, Normal, and 1 is for Lung opacity.In the stage_2_detailed_class_info.csv file,

there are two columns patientId and class column that describe the three conditions of lungs (See Figure 3). Have combined both these datasets to create a new dataframe.

| | patientId | x | y | width | height | Target | class |
|---|--------------------------------------|-------|-------|-------|--------|--------|------------------------------|
| 0 | 0004cfab-14fd-4e49-80ba-63a80b6bdd6 | NaN | NaN | NaN | NaN | 0 | No Lung Opacity / Not Normal |
| 1 | 00313ee0-9eaa-42f4-b0ab-c148ed3241cd | NaN | NaN | NaN | NaN | 0 | No Lung Opacity / Not Normal |
| 2 | 00322d4d-1c29-4943-afc9-b6754be640eb | NaN | NaN | NaN | NaN | 0 | No Lung Opacity / Not Normal |
| 3 | 003d8fa0-6bf1-40ed-b54c-ac657f8495c5 | NaN | NaN | NaN | NaN | 0 | Normal |
| 4 | 00436515-870c-4b36-a041-de91049b9ab4 | 264.0 | 152.0 | 213.0 | 379.0 | 1 | Lung Opacity |

Figure 3- Class data

The frequency of patients in each class and their respective percentages are shown in Figure 4.

23.5 percent of the patients are Normal, and the remaining are Not Normal and Lung opacity.

```
'''define a function to change the type from object / catagorical to int '''
def changeType(x):
    if x == 'No Lung Opacity / Not Normal':
        return 2
    elif x == 'Normal':
        return 0
    elif x == 'Lung Opacity':
        return 1
```

```
#find the value count for the df
printwithinseparator(mergedf['class'].value_counts(),'merge dataframe with unique class and count'
# mergedf['class'].value_counts()

===== merge dataframe with unique class and count =====
2    11821
1     9555
0     8851
Name: class, dtype: int64
```

Figure 4 - Count of patients in each class

The bounding box dataset has missing values in x, y, height, and width column. (See Figure 5).

```
#by data check info we can check is there any missing data or now so we can eliminate or manipulate furt.
data_check.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 9555 entries, 4 to 37627
Data columns (total 7 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   patientId   9555 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        9555 non-null   int64  
 6   class         9555 non-null   int64  
dtypes: float64(4), int64(2), object(1)
memory usage: 597.2+ KB
```

Figure 5- Missing Values

So, the remaining 16957 are the positive means Lung Opacity case. Figure 6 shows the count plot of the three classes.

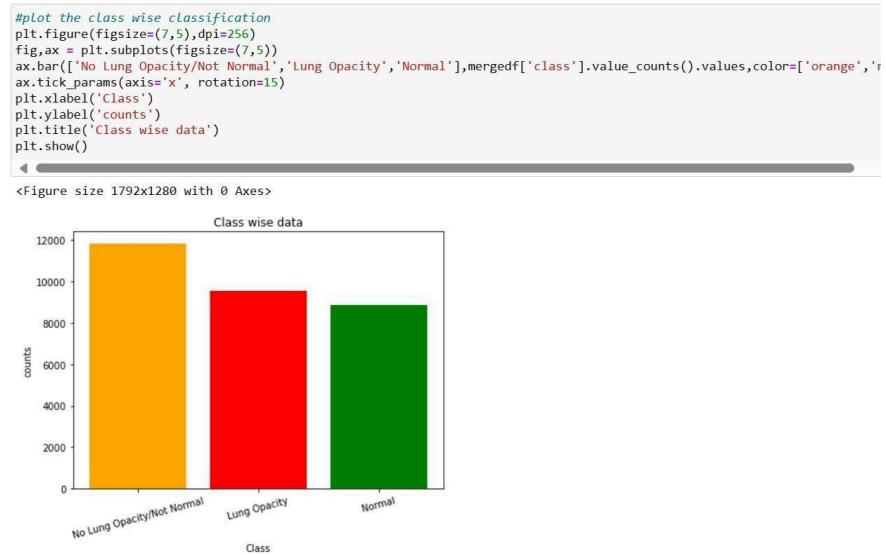


Figure 6 - Bar diagram of patients in each classes

There are 26684 training images and 3000 test images. Visualizations of the few samples are shown in Figure 7.

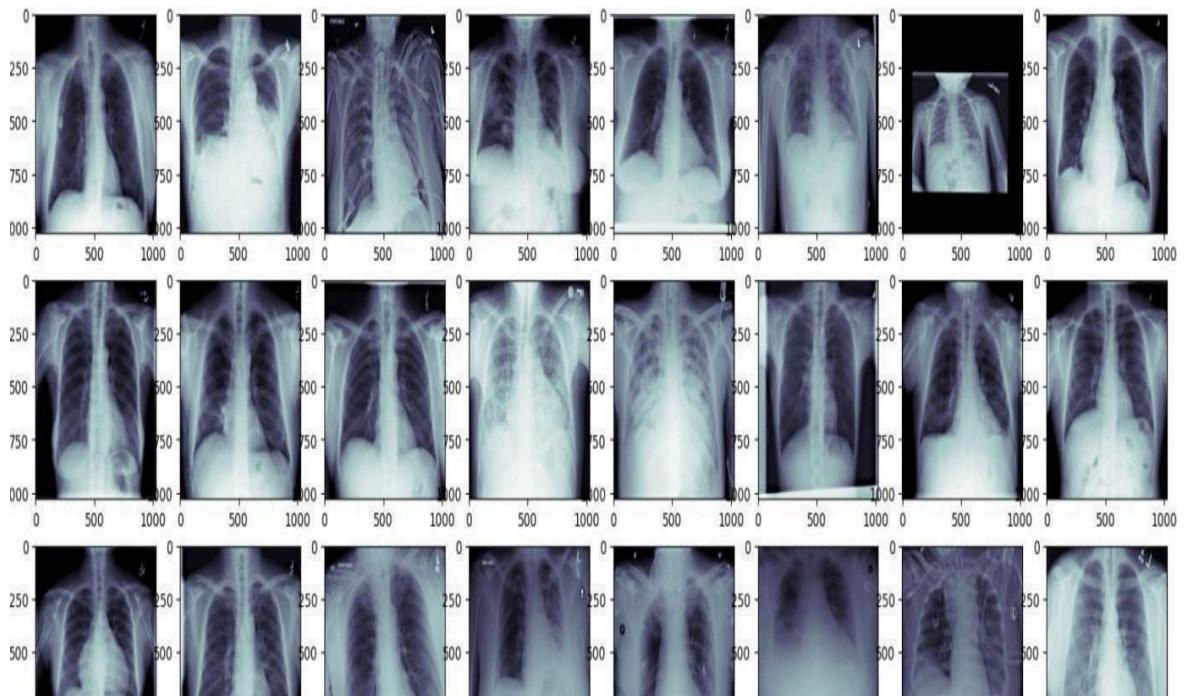


Figure 7- Sample visuals of chest X-Ray images

The images, along with the bounding box, is presented in Figure 8. The figure indicates that some images have more than one bounding box, whereas some do not even have one.

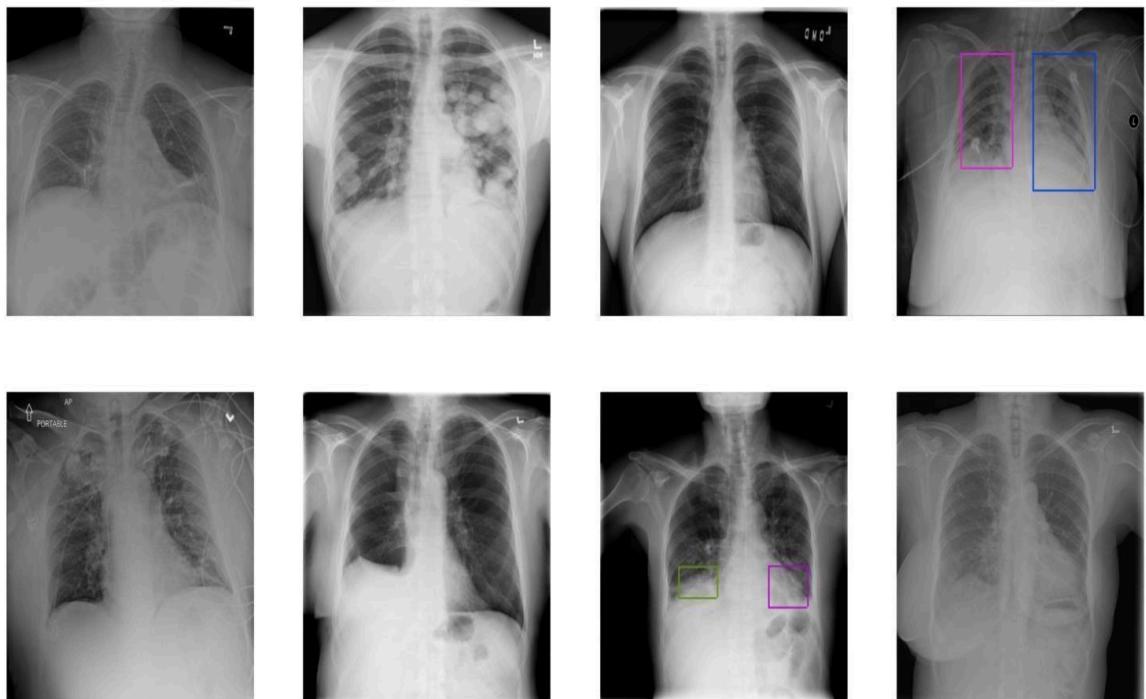


Figure 8- Sample visuals of chest X-Ray images along with Bounding Boxes

As we can determine from the above visualizations, the task in hand is a regression problem.

There is a need for building a feasible model that can regress the bounding box in the images. Moreover, there is an extra class that is Not normal/ No lung opacity. This class shows there is an anomaly in the lungs, which can be easily misread as pneumonia. So, there is a need to examine that class a little more briefly. A visualization showing all three classes together is shown Figure 9.



Figure 9 - Sample visuals of chest X-Ray images along with Bounding Boxes for the three classes.

A bar plot of the target classes is shown in Figure 10.

```
1 sns.set(style="whitegrid")
2
3 # Create the countplot
4 plt.figure(figsize=(10, 6))
5 sns.countplot(x='Target', hue='class', data=info_df)
6 plt.title('Relationship Between Class and Target')
7 plt.xlabel('Target')
8 plt.ylabel('Count')
9 plt.legend(title='Class')
10 plt.xticks(rotation=45)
11 plt.tight_layout()
12 plt.show()
13
```

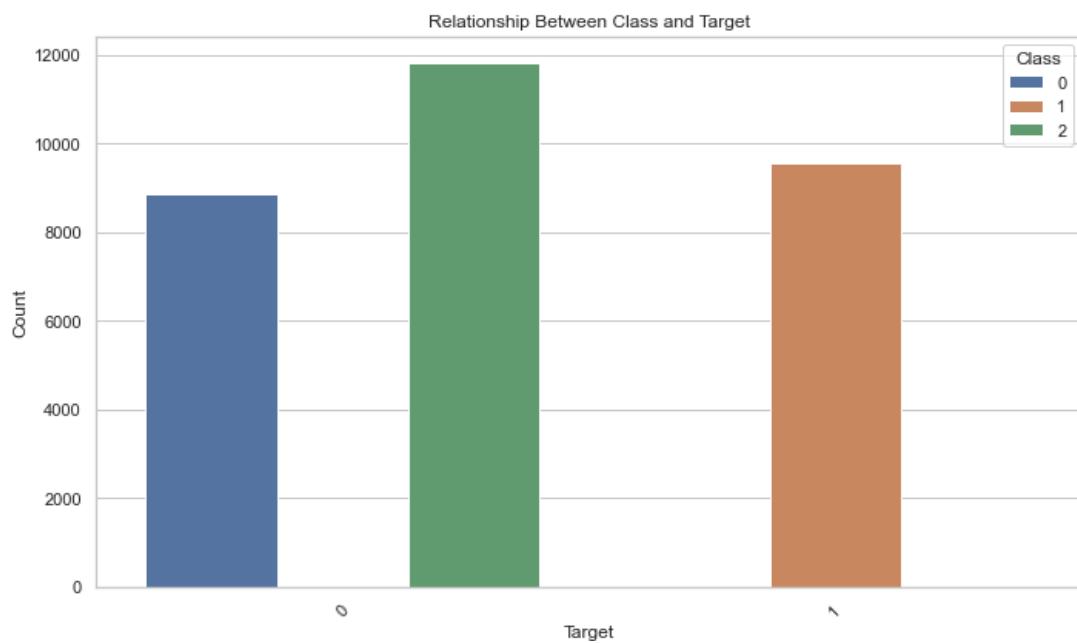


Figure 10 - Bar diagram representing relationship between class and target

The information regarding the patients is available in the metadata of the Dicom images. Visualizations of that data may give a better understanding of the pneumonia disease itself. Figure 11 shows the data frame of the extracted dicom data.

```

#read images dicom and get the info
first_dicom_file = images_df[0]
first_dicom_file_path = os.path.join('train_images\\stage_2_train_images\\', first_dicom_file)
dcm = pdcm.dcmread(first_dicom_file_path)
dcm.fix_meta_info

<bound method Dataset.fix_meta_info of Dataset.file_meta -----
(0002, 0000) File Meta Information Group Length UL: 202
(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: 1.2.276.0.7230010.3.1.4.8323329.28530.1517874485.775526
(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: 1.2.276.0.7230010.3.1.4.8323329.28530.1517874485.775526
(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: PA'
(0010, 0010) Patient's Name PN: '0004cfab-14fd-4e49-80ba-63a80b6bdd6'
(0010, 0020) Patient ID LO: '0004cfab-14fd-4e49-80ba-63a80b6bdd6'
(0010, 0030) Patient's Birth Date DA: ''
(0010, 0040) Patient's Sex CS: 'F'
(0010, 1010) Patient's Age AS: '51'

```

Figure 11 - Data Frame of the extracted dicom data

Gender is one of the variables present in the data which can be explored. We can infer that the dataset has more male examples from the below images than the female examples. In this case

there are more men with pneumonia, around 4800 compared to around 3300 women with pneumonia (See Figure 12)



Figure 12 – Bar Diagram representing the gender of the patients

The abnormality in the lungs is very high in men compared to women. There is a massive difference in the Not normal/ No lung opacity class between males and females (See Figure 13).

Have resized the images to 512 size and then created a dataset info_df which contains more columns representing additional fields from DICOM images

```
info_df = mergedf.copy()
total_images = info_df['patientId'].unique()

info_df['AGE'] = 0
info_df['SEX'] = 0
info_df['ViewPosition'] = ''
info_df['BodyPart'] = ''
info_df['glcm_contrast'] = ''
info_df['glcm_homogeneity'] = ''
info_df['glcm_energy'] = ''
info_df['glcm_correlation'] = ''
```

Figure 13 – Extracting features from DICO

The dataframe has following info (see Figure 14)

```
1 glcmcol = [ 'glcm_contrast', 'glcm_homogeneity', 'glcm_energy', 'glcm_correlation']
2 for i in glcmcol:
3     info_df[i] = pd.to_numeric(info_df[i])
4
5 info_df.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 30227 entries, 0 to 37627
Data columns (total 15 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  
 6   class            30227 non-null   int64  
 7   AGE              30227 non-null   object 
 8   SEX              30227 non-null   int64  
 9   ViewPosition     30227 non-null   object 
 10  BodyPart         30227 non-null   object 
 11  glcm_contrast    30227 non-null   float64
 12  glcm_homogeneity 30227 non-null   float64
 13  glcm_energy      30227 non-null   float64
 14  glcm_correlation 30227 non-null   float64
dtypes: float64(8), int64(3), object(4)
memory usage: 4.7+ MB
```

Figure 14 – New dataset with features from DICOM images.

The pairplot is used to identify relationships between variables in our dataset. We aim to understand how bounding box coordinates, patient demographics (age, gender), and texture features extracted from images relate to the presence or absence of pneumonia. This visualization helps us uncover any patterns or correlations that may exist among these factors. (See Figure 15)

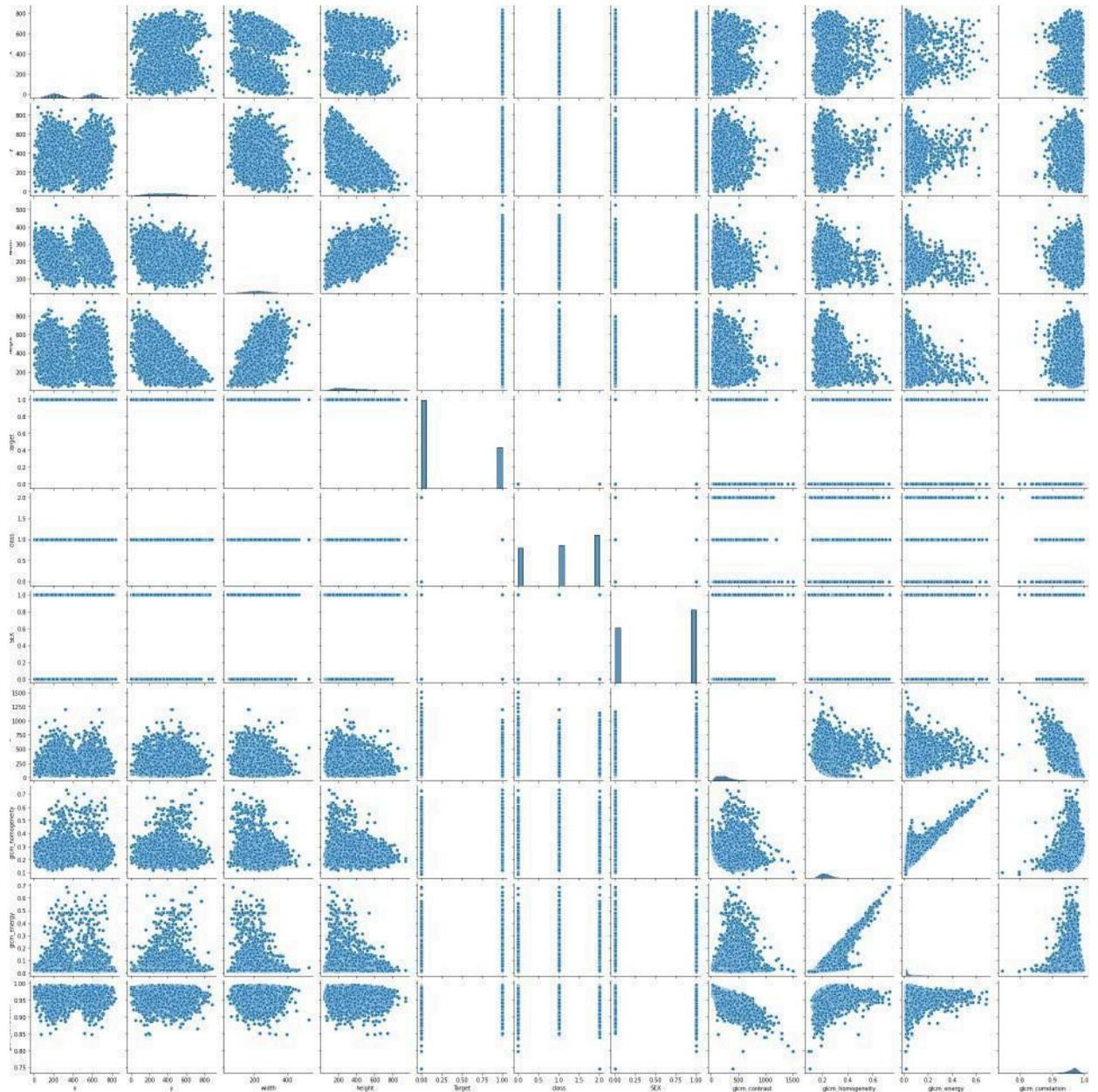


Figure 15 – Pairplot for DICOM features.

Tried boxplots to find correlation of these fields with Target.

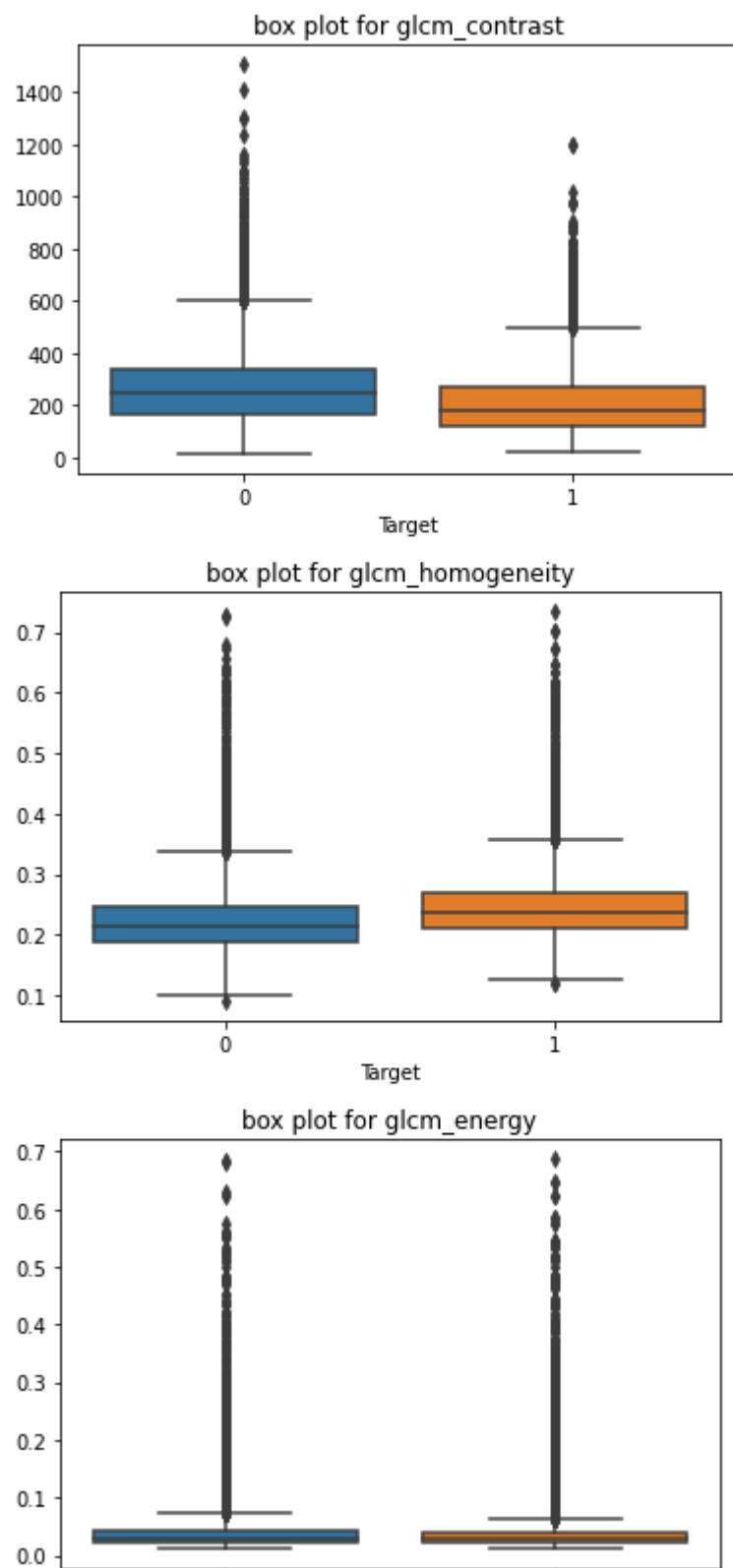


Figure 16 – Boxplots for DICOM features

Also created a heatmap which show no significant correlation between the features(see figure 17)

<AxesSubplot:>

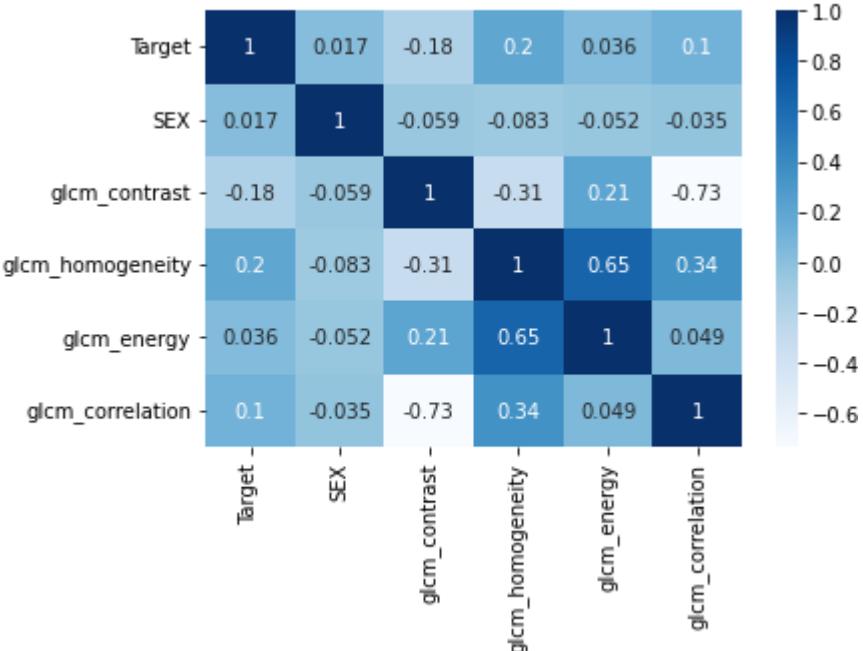


Figure 17 – Heatmap for DICOM extracted features

```
1 target_counts = info_df.query('Target ==1')['agecat'].value_counts()  
2  
3 # Plotting the distribution as a pie chart  
4 plt.figure(figsize=(8, 8))  
5 plt.pie(target_counts, labels=target_counts.index, autopct='%1.1f%%')  
6 plt.title('Distribution of Targets')  
7 plt.show()
```

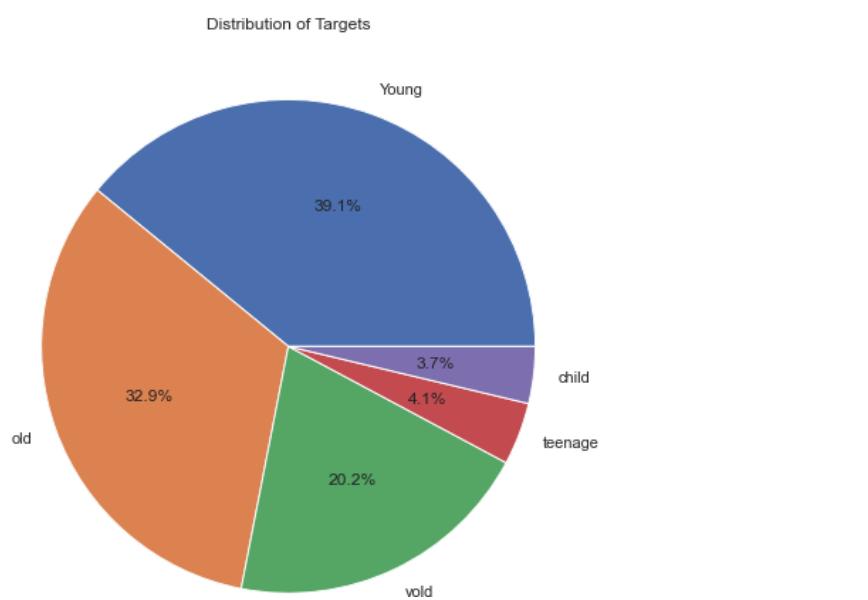


Figure 18 – Distribution based on age categories

Based on the age categories we found that old and young people have highest cases of Pneumonia.

The purpose of these scatter plots is to visually explore the relationship between the bounding box coordinates (x , y) and their corresponding width and height for images where pneumonia is present (Target = 1). By plotting these variables against each other, we can observe any patterns or correlations that may exist. This analysis helps in understanding the distribution and positioning of pneumonia regions within the chest X-ray images. (See Figure 19)

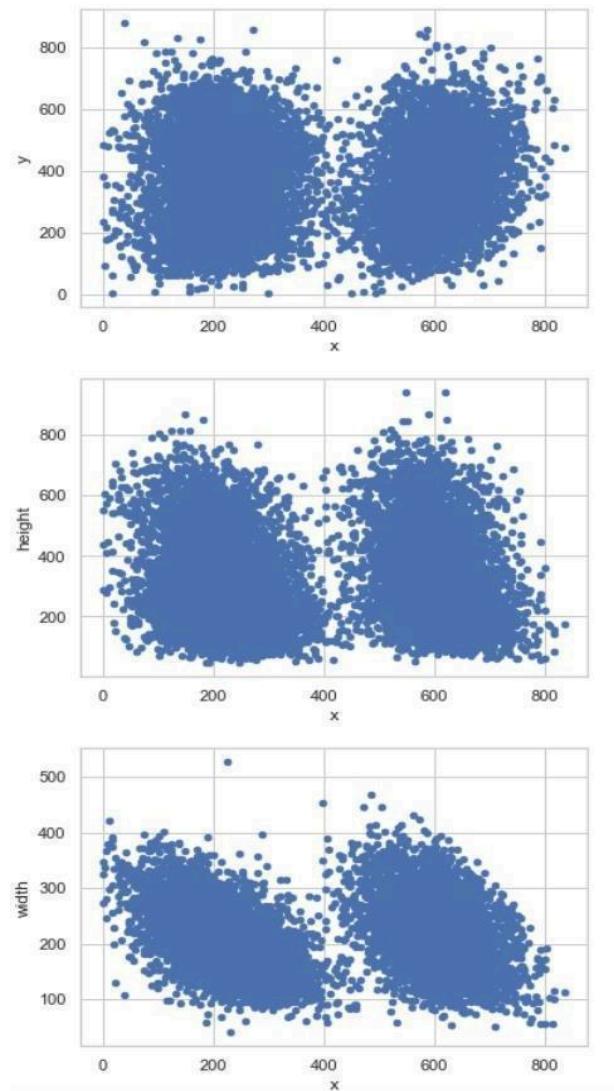


Figure 19 – Scatterplot

STEP-BY-STEP WALK THROUGH THE SOLUTION

Based on the findings from exploratory data analysis and problem statement, it is evident that the model should be a bounding box regressor that can identify the lung opacities, in turn predicting pneumonia. The model should have the ability to localize and identify the opacities. So, based on that, we came up with the following models.

CNN MODEL:

This model consists of a series of residual blocks in the middle with downsampling then followed by output block, which leads to upsampling.

Approach

The approach involves training a Convolutional Neural Network (CNN) to automatically detect pneumonia from chest X-ray images. The dataset is split into training and validation sets, and image augmentation techniques are applied to prepare the data for training. The CNN architecture is designed to extract features from chest X-ray images and classify them into three categories: "No Lung Opacity / Not Normal", "Normal", and "Lung Opacity".

Network

We're leveraging a Convolutional Neural Network (CNN) for this task. A CNN is a sophisticated computational framework composed of numerous layers designed to analyze X-ray images systematically. Its primary function is to identify key features within the images, such as dark spots which indicate pneumonia. By aggregating these features, the network determines whether the X-ray exhibits signs of a healthy condition or pneumonia. This helps medical professionals in correctly evaluating X-rays and devising optimal treatment plans for patients.

Results

- The model was trained for 25 epochs on a dataset comprising 2400 images for training and 600 images for validation which were categorized into 3 classes.
- Throughout training, the model showed gradual improvement in accuracy, reaching a peak training accuracy of ~65% and a peak validation accuracy of ~59% by the 8th epoch.
- Despite the improvement in accuracy, the model's performance plateaued at around 50% validation accuracy, indicating the need for further optimization or adjustments.
- The training process was halted after the 5th epoch as there was no significant improvement in validation accuracy, suggesting that the model may have reached its

learning capacity with the current configuration.

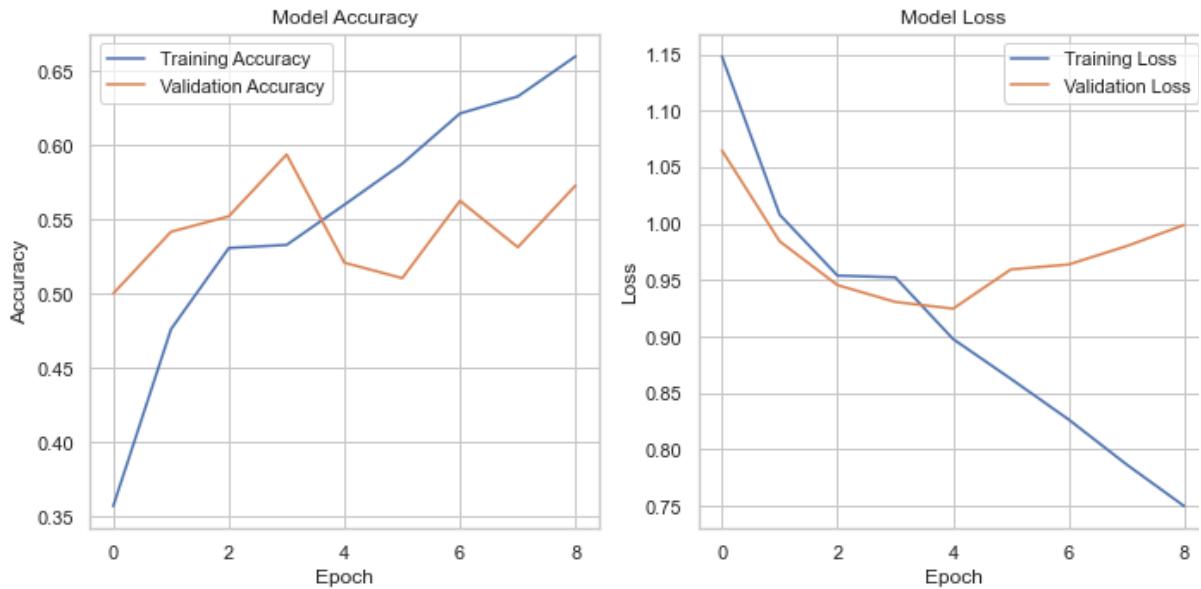


Figure 19-Training and Validation Performance of CNN Model.

Chest X-ray Classification Model with Enhanced Convolutional Layers

This model utilizes additional convolutional layers and dropout regularization compared to the previous version, aiming to improve accuracy in classifying chest X-ray images into three categories.

Approach

The new model adopts a revised architecture with fewer convolutional layers and an additional dense layer compared to the previous model.s.

Network

Compared to the previous model, which consisted of four convolutional layers, the new model integrates three convolutional layers followed by max-pooling for feature extraction. Additionally, it includes an extra dense layer for classification.

Results

The model training results indicate an evolution from the initial accuracy of around 47% to a peak validation accuracy of approximately 65% over 30 epochs. However, the model's performance oscillates during training, suggesting that further optimization may be necessary to achieve more consistent results.

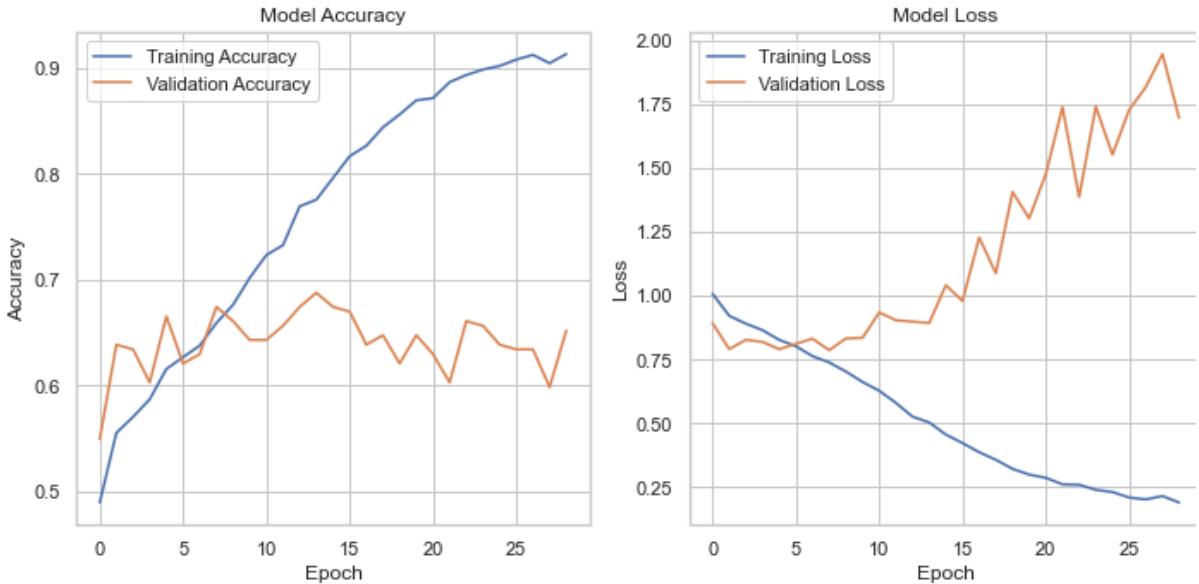


Figure 20-Training and Validation Accuracy Comparison of Enhanced Convolutional Model

Enhanced Convolutional Model with Increased Dataset Size

The model utilizes an expanded dataset containing 3000 samples in each of three classes. It employs a convolutional neural network architecture with three convolutional layers followed by max-pooling layers, aiming to classify medical images accurately.

Approach

The dataset is augmented by sampling 3000 instances from each class. Data preprocessing includes converting image labels to strings and appending '.jpg' to file paths. The dataset is split into training and validation sets. ImageDataGenerator is used for data augmentation and rescaling. A CNN model is constructed with convolutional and max-pooling layers, dropout layers, and softmax activation. Model is compiled with Adam optimizer and categorical cross-entropy loss. Callbacks are used for monitoring validation accuracy and saving the best model.

Network

The CNN architecture includes three convolutional layers with (3,3) filter sizes, followed by max-pooling layers (2,2). ReLU activation is used in convolutional layers. A Flatten layer is added to flatten the feature maps. Two dense layers with 32 and 16 units, respectively, use ReLU activation. Dropout layer (dropout rate=0.5) prevents overfitting. Output layer with softmax activation has three units for classification probabilities.

Results

The model reached its best performance with a validation accuracy of nearly 61% after training

for 30 epochs. This shows a bit of improvement compared to the last model. However, it seems like the model couldn't get much better after reaching this point, suggesting it might be too focused on the training data and not able to perform as well on new, unseen data. So, we might need to try different ways to make the model better or find a different approach altogether.

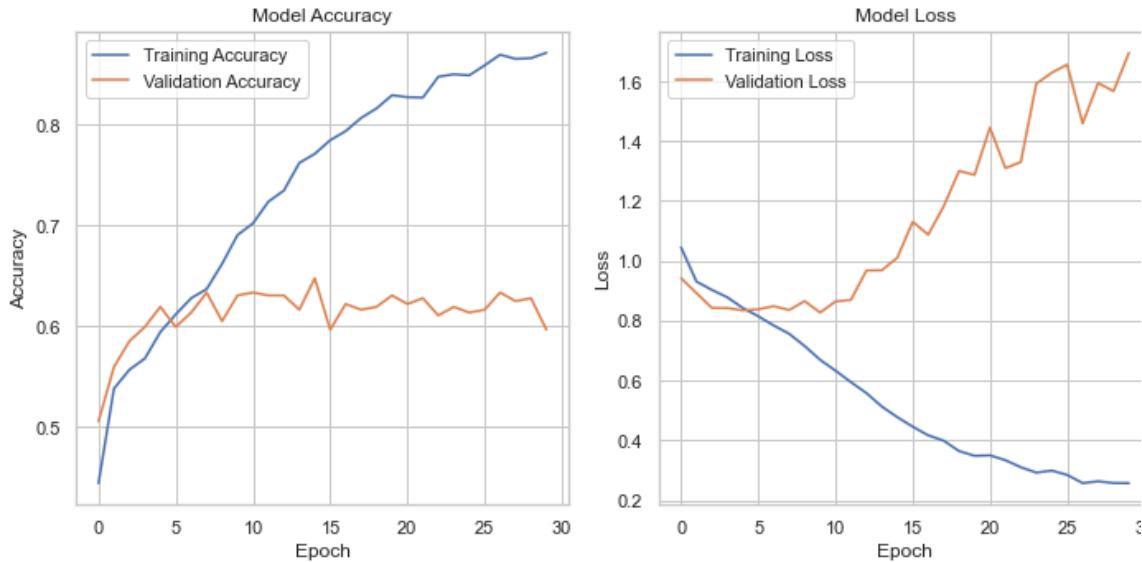


Figure 21- Training and Validation Accuracy Evolution

Upon evaluating the trained model, several key performance metrics were analyzed to gauge its effectiveness in classifying chest X-ray images. These metrics include Precision, Recall, F1 Score, and Accuracy.

- Precision measures the accuracy of the positive predictions made by the model. In this case, the precision is approximately 0.34, indicating that around 34% of the predicted positive cases were correct.

```

12/12 [=====] - 1s 52ms/step - loss: 1.0112 - accuracy: 0.6417
Validation Accuracy: 0.6416666507720947
Precision: 0.3445913141122871
Recall: 0.35
F1 Score: 0.345327288207748

```

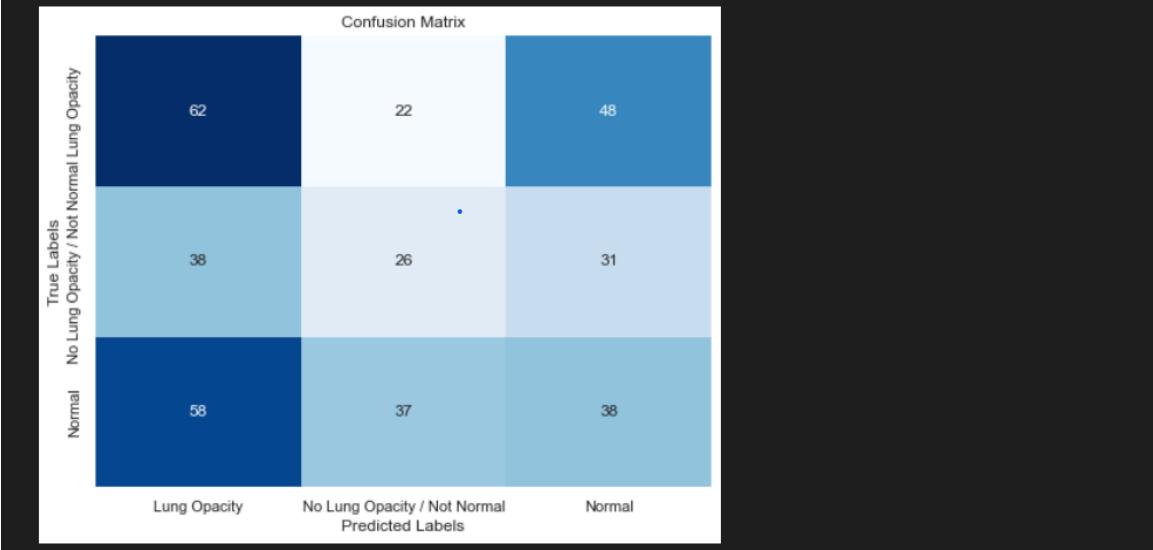


Figure 20- Confusion matrix obtained for the CNN Model

- Recall, also known as sensitivity, measures the ability of the model to correctly identify positive instances from all actual positive instances. The recall here is approximately 0.35, indicating that the model identified around 35% of all actual positive cases.
- The F1 score is the harmonic mean of precision and recall, providing a single metric to assess the model's performance. The F1 score achieved is approximately 0.34, indicating a balanced performance between precision and recall.
- The Accuracy is 64%

-Confusion matrix

The confusion matrix provides a detailed breakdown of the model's performance across different classes. Each row represents the actual class, while each column represents the predicted class. The values in the matrix represent the count of observations. Figure 20 shows the confusion matrix obtained for the CNN model. The values illustrate the model's performance in correctly classifying instances of each class and identify misclassifications. For instance, the model appears to have relatively higher accuracy in predicting the "Normal" class compared to the other classes, while it struggles more with the "No Lung Opacity / Not Normal" class. Further analysis and potential adjustments may be necessary to improve performance, particularly for classes with lower precision and recall.

Summary:

Three CNN models were trained for medical image classification: the baseline CNN, CNN with 256x256 image resolution, and CNN with additional data augmentation. The baseline model achieved an accuracy of 59.8% on the validation set, while the model with more numbers of images reached 60.9% accuracy. However, both models showed signs of overfitting. The third model, with data augmentation, yielded an accuracy of ~65% but displayed improved generalization.

Challenges faced:

we have tried to run the model with all the images we have in the dataset but stuck and backstopped because of a lack of resources to GPU and computation power so tried with chunks of data and fine-tuned data with the balanced dataset. Still, there are many challenges to work to improve the performance by using other current libraries like CUDF and CUML for parallel processing.

Scope of Improvement:

To further enhance model performance, several optimization strategies can be explored. Firstly, fine-tuning hyperparameters such as learning rate, batch size, and dropout rate may mitigate overfitting and improve convergence. Secondly, experimenting with advanced CNN architectures like ResNet, DenseNet, VGG, Mobile Net, or Inception could leverage their deeper structures and skip connections to enhance feature extraction and classification accuracy. Finally, exploring transfer learning techniques by pre-training models on larger medical image datasets before fine-tuning on the target dataset may also lead to improved performance.

CLOSING REFLECTIONS

As our approach resulted in reaching the benchmark, it can be illustrated that our approach to the problem is experimental. The lack of domain knowledge in the initial phase of the project is slow and inefficient. We studied the dataset with significantly less idea about the inner workings of pneumonia or opacities. So, we concur that a domain expert is essential for a more feasible and efficient solution.

We conclude that even though there have been substantial deep learning advances in radiology and medicine, we need better models and strategies that are majorly dedicated to those fields as the amount of data is vast. The margin of error allowed is very small or sometimes none.