****

**RSNA PNEUMONIA DETECTION**

**Team:**

|  |
| --- |
| **Aaron Pereira**  **Sahrash Shaqeeb**  **Dhone Sureendra**  **Vaishakh Kombilath**  **Project Summery**  In this capstone project, the goal is to build a pneumonia detection system, to locate the position of inflammation in an image. Tissues with sparse material, such as lungs which are full of air, do not absorb the X-rays and appear black in the image. Dense tissues such as bones absorb X-rays and appear white in the image. While we are theoretically detecting “lung capacities”, there are lung capacities that are not pneumonia related. In the data, some of these are labeled “Not Normal No Lung Opacity”. This extra third class indicates that while pneumonia was determined not to be present, there was nonetheless some type of abnormality on the image and oftentimes this finding may mimic the appearance of true pneumonia. Dicom original images: - Medical images are stored in a special format called DICOM files (\*.dcm). They contain a combination of header meta data as well as underlying raw image arrays for pixel data. |
|  |

**Problem Statement**

In this capstone project, the goal is to build a pneumonia detection system, to locate the position of inflammation in an image. Tissues with sparse material, such as lungs which are full of air, do not absorb the X-rays and appear black in the image. Dense tissues such as bones absorb X-rays and appear white in the image. While we are theoretically detecting “lung capacities”, there are lung capacities that are not pneumonia related. In the data, some of these are labeled “**Not Normal No Lung Opacity**”. This extra third class indicates that while pneumonia was determined not to be present, there was nonetheless some type of abnormality on the image and oftentimes this finding may mimic the appearance of true pneumonia.

**Objectives**

* Learn to how to do build an Object Detection Model
* Use transfer learning to fine-tune a model.
* Learn to set the optimizer, loss functions, epochs, learning rate, batch size, check pointing, early stopping etc
* Read different research papers of given domain to obtain the knowledge of advanced models for the given problem.

**Methodologies**

1. Data set preparation

* Cleaning
* Visualization
* Statistical analysis

1. Image Augmentation

* Image conversion .dcm to .png
* Re sizing as per the model preprocessing requirement

1. Model Building

* Model initialization
* Compiling/Fit the data

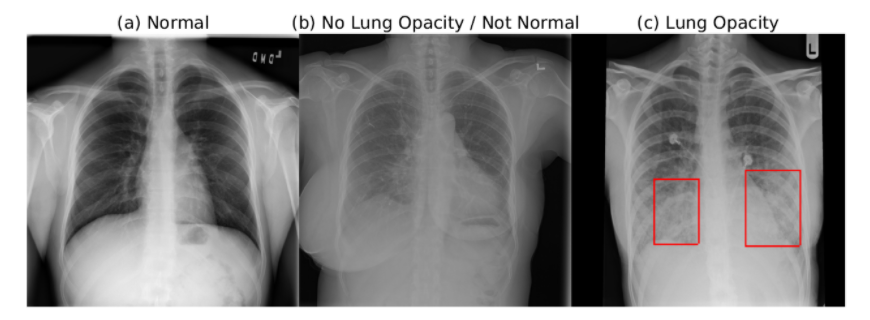
1. Testing accuracy

* Classification report
* Model performance graph

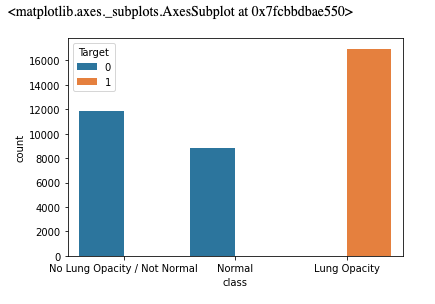
**Dataset/ EDA**

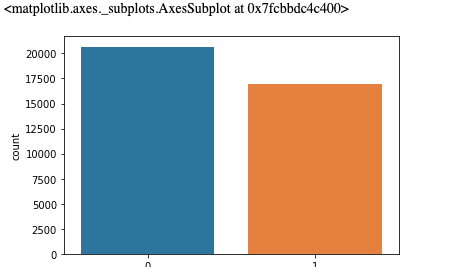
The labeled data-set of the chest X-Ray (CXR) images and patients meta data was publicly provided for the challenge by the US National Institutes of Health Clinical Center. The data-set is available on kaggle platform.

The database comprises frontal-view X-ray images from 26684 unique patients. Each image is labeled with one of three different classes from the associated radiological reports: ”Normal”, ”No Lung Opacity / Not Normal”, ”Lung Opacity”. Fig. 1 shows examples of all three classes CXRs labeled with bounding boxes for unhealthy patients.

****

Since we have 3 classes, the distribution among total data is as shown below.



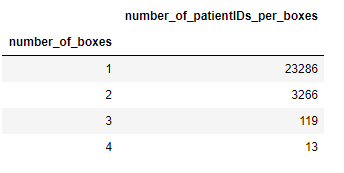


**Checking Distribution of Target**

From the CSV data, the target column gives the information of the whether the patient is having pneumonia positive or not by **1**&**0.** The distribution of the target is as shown below. Total positive cases are 32% and Negative cases are 68%.

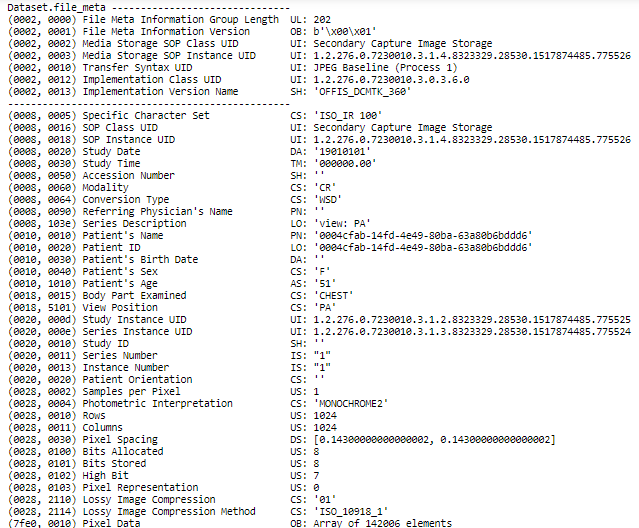
**Initial Observations**

* Null values are present only with bounding box data I the given CSV.
* All the bounding box null values are associated with target **0**
* Each patient ID is associated with a single class or target
* Many of the patient id’s are associated with more than one bounding boxes



****

* All positive cases are associated with target **1** only.
* The X-ray images are in .dcm format with a resolution of 1024.
* We have total 30227, in that missing value samples 20672 and 9555.
* We have unique patient id’s 26684 and we have duplicate patient id’s of 3543.

**Sample patient information from the .dcm image is as shown below**

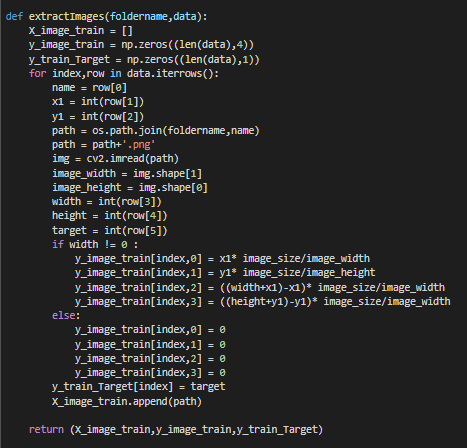
**Steps Followed in Data Preparation**

* Understanding the data with a brief on train/test labels and respective class info
* Look at the first five rows of both the .csv files(train and test).
* Identify how are classes and target distributed
* Check the number of patients with 1, 2, ... bounding boxes
* Read and extract meta data from dicom files
* Perform analysis on some of the features from dicom files
* Check some random images from the training dataset
* Draw insights from the data at various stages of EDA
* Visualize some random masks generated

From the dataset, we have classification and regression statement. Whereas the classification part comes with predicting pneumonia positive or negative and the regression part has to predict the area which opacity has found and draw the bounding box.

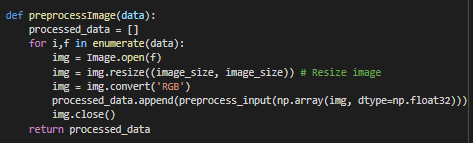
**Image Extraction**

Image extraction involves saving image path in input training variable along with making bounding dependency variable & target variable. We handled Null/Nan bounding box variable as zero.



**Image Preprocessing**

Pre-processing involves scaling the image to desired size for the selected model. We used PIL package to read, re-size and converting image to RGB(to make 3 channel).

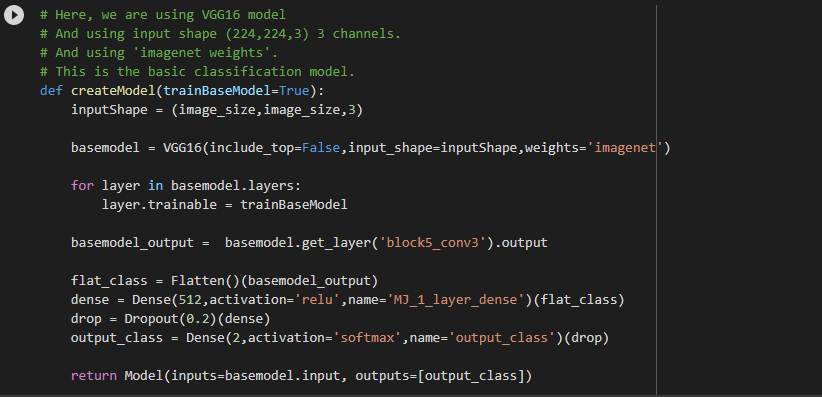


**Model Selection**

We have prepared two basic models using Resnet50 and VGG16 architecture with pre-trained “imagenet” weights. Here we are only training top layers which is defined by our self. The model has an input shape of images width, height and channels 224, 224, 3 respectively.

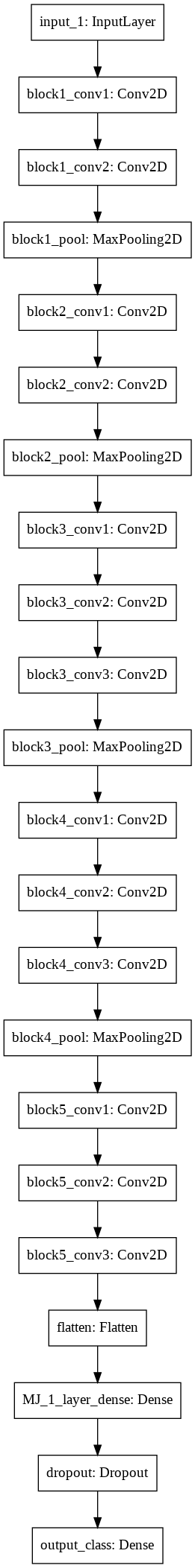
The model is fit on 10000 input samples & validation done on 2000 samples.

**VGG16 - CNN Model**

**Creating Model**

**Model Summery**

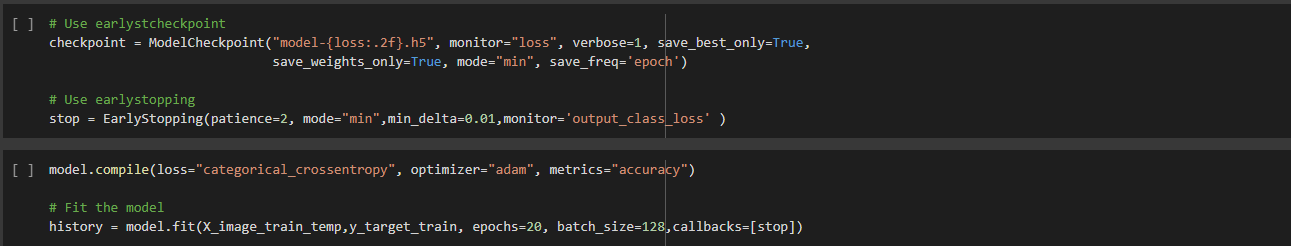
****

**VGG16 Flowchart**

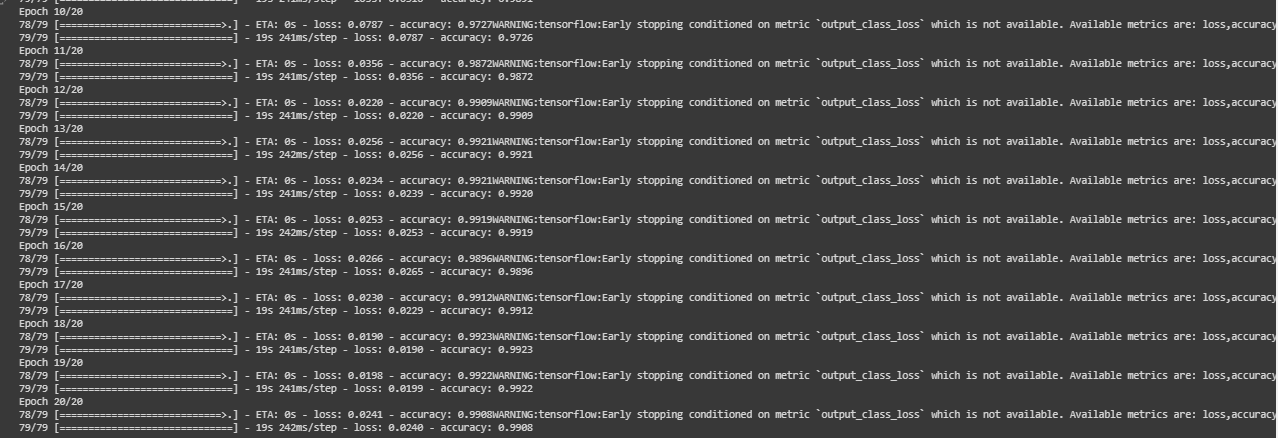
**Model Performance**

The model is fit on 10000 input samples & validation done on 2000 samples.

* + - * Defining loss function & fitting the model

****

**Model Performance Visualization**

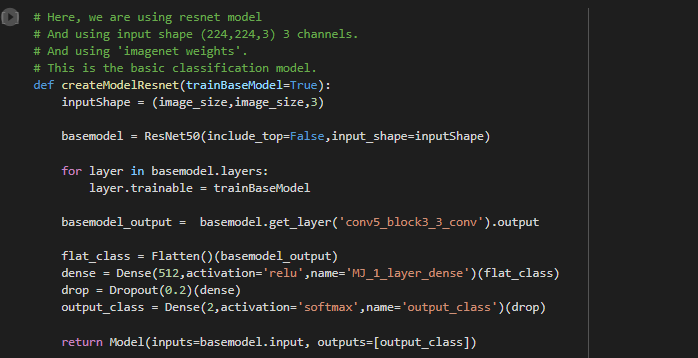
****

**Model Performance Visualization**

****

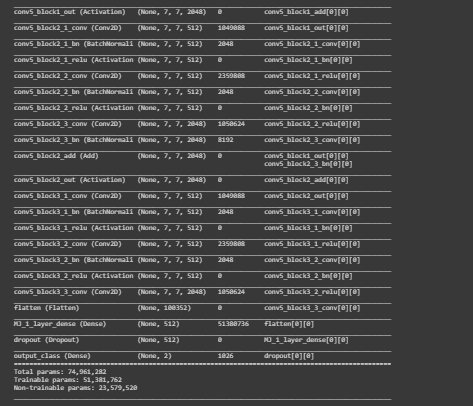
**RESNET 50 - CNN Model**

**Creating Model**

****

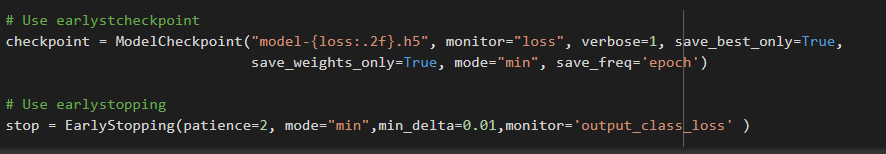
**Model Summery**

* Final blocks shown



**Model Performance**

The model is fit on 10000 input samples & validation done on 2000 samples.



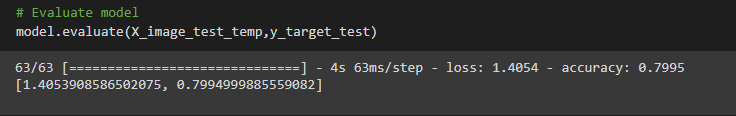
****

**Model Performance Visualization**

****

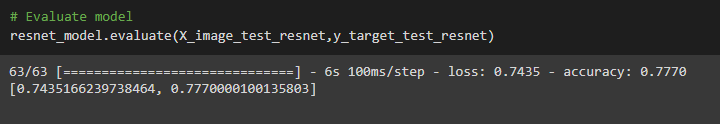
**VGG16 vs. ResNet50(Accuracy Comparison)**

**VGG16 Model Evaluation**

****

Output classification accuracy: 80%

**RESNET50 Model Evaluation**

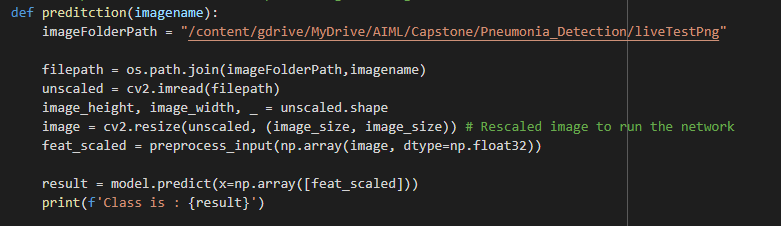
****

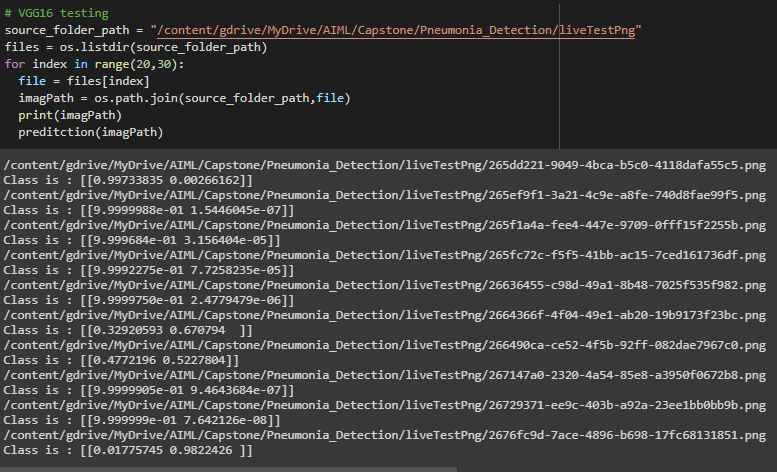
Output classification accuracy: 77%

**Selecting model for final prediction**

From the above comparison it is clear that VGG16 gives better accuracy in classification compared to RESNET50. So we are selecting VGG16 for the final prediction.

**VGG-16 Prediction**





**Conclusion**

We have analyzed all 30k images and we got to know that duplicated patient ids and we have considered this as different samples for the model input. In model building process we choose to go for transfer learning. So in that we choose two models VGG16 and ResNet50 based on the performance compared we have found VGG16 is better than that ResNet50.