CAPSTONE PROJECT - COMPUTER VISION

TOPIC: PNEUMONIA DETECTION USING CHEST X-RAYS



TABLE OF CONTENTS

1. OVERVIEW 03
2. PROBLEM STATEMENT, DATA, AND FINDINGS 04
3. EXPLORATORY DATA ANALYSIS 05

3a. DATASET INFORMATION 05

3b. DATA MERGING 06

3c. CLASS AND TARGET INFORMATION 07

3d. IMBALANCE CLASS PROBLEM 07

3e. EDA ON DICOM IMAGES DATASET 09

3f. DICOM IMAGES WITH BOUNDING BOX LOCATING LUNG OPACITY 09

1. DATA PRE-PROCESSING 10
2. MODEL BUILDING AND EVALUATION 12

5a. TRAINING AND TESTING DATA SPLITS 12

5b. CONVOLUTIONAL NEURAL NETWORK (BASE MODEL) 12

5c. TRANSFER LEARNING MODEL 13

1. TEST PREDICTION AND FINAL SUBMISSION EXCEL SHEET 15

7a. PREDICTING RESULT 15

7b. SUBMISSION 16

1. LEARNING AND IMPROVEMENTS 16

OVERVIEW:

Pneumonia is an infection in one or both lungs. Bacteria, viruses, and fungi cause it. The infection causes inflammation in the air sacs in your lungs, which are called alveoli.

Pneumonia accounts for over 15% of all deaths of children under 5 years old internationally. In 2015, 920,000 children under the age of 5 died from the disease.

While common, accurately diagnosing pneumonia is a tall order. It requires review of a chest radiograph (CXR) by highly trained specialists and confirmation through clinical history, vital signs and laboratory exams.

Pneumonia usually manifests as an area or areas of increased opacity on CXR. 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.

CXRs are the most commonly performed diagnostic imaging study. A number of factors such as positioning of the patient and depth of inspiration can alter the appearance of the CXR, complicating interpretation further.

Source: RSNA Pneumonia Detection Challenge from Kaggle ( <https://www.kaggle.com/c/rsna-pneumonia-detection-challenge/overview> )

PROBLEM STATEMENT, DATA, AND FINDINGS

**PROBLEM STATEMENT:**

* The goal is to build a pneumonia detection system to locate the position of inflammation in an image.
* While we are theoretically detecting “lung opacities”, there are lung opacities that are not pneumonia related. some of these are labelled “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

**DATA AND FINDINGS:**

**Data Source:**

RSNA Pneumonia Detection Challenge by Kaggle (<https://www.kaggle.com/c/rsna-pneumonia-detection-challenge/data>)

The data is organized in folders:

* The training data is provided as a set of patient Ids and bounding boxes. Bounding boxes are defined as follows: x-min, y-min, width, height
* There is also target column “ Target “ , indicating pneumonia or non-pneumonia.
* There may be multiple rows per patient Id
* All provided images are in DICOM format

**stage\_2\_detailed\_class\_info.csv:**

In **class detailed info** dataset are given the detailed information about the type of positive or negative class associated with a certain patient.

**stage\_2\_train\_labels.csv**:

In **train labels** dataset are given the patient ID and the window (x min, y min, width and height of the) containing evidence of pneumonia.

**stage\_2\_train\_images.zip**:

The directory containing 26000 training set raw image (DICOM) files

**stage\_2\_test\_images.zip**:

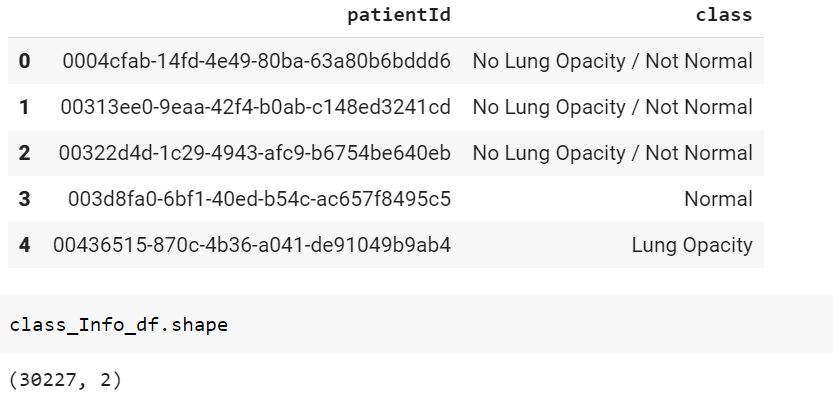
The directory containing 3000 testing set raw image (DICOM) files

EXPLORATORY DATA ANALYSIS

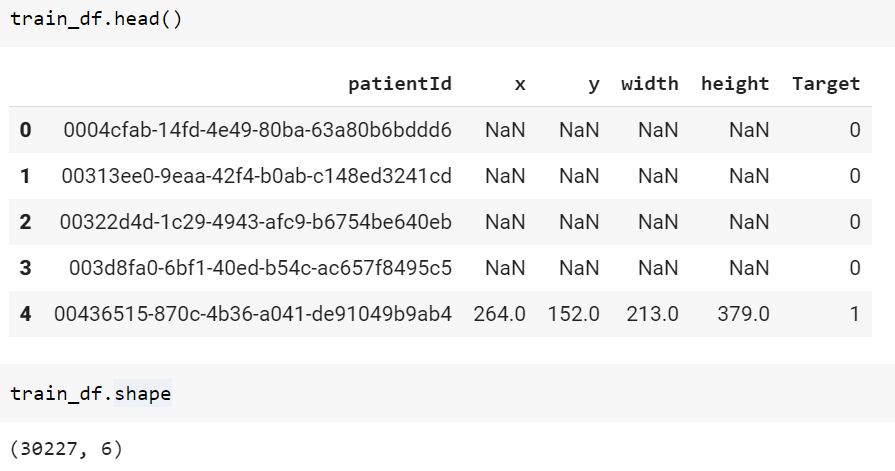
**DATASET INFORMATION**:

Stage\_2\_detailed\_class\_info.csv dataset:

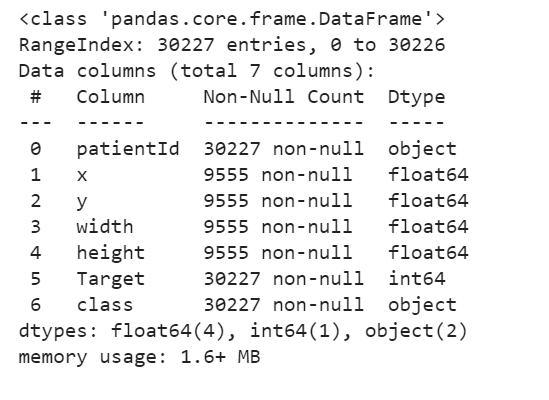
Class info csv contains 30227 rows with 2 columns patient Id and class



Structure of stage\_2\_train\_labels.csv dataset: train\_labels.csv contains 30227 rows with columns



DATA FIELD AND TYPES in the train\_labels.csv:



**patientId** – Patient Id corresponds to identification of images.

**X** - the upper-left x coordinate of the bounding box.

**Y** - the upper-left y coordinate of the bounding box.

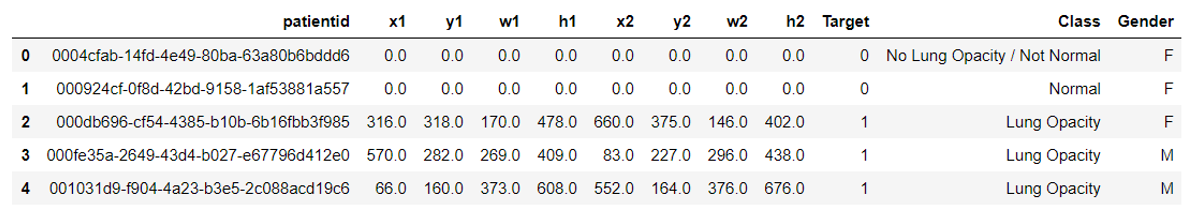
**Width** - the width of the bounding box.

**Height** - the height of the bounding box.

**Target** - the binary Target, indicating whether this sample has evidence of pneumonia

Each row in the CSV file contains a patient Id (one unique value per patient), a target (either 0 or 1 for absence or presence of pneumonia, respectively) and the corresponding abnormality bounding box defined by the upper-left hand corner (x, y) coordinate and its corresponding width and height.

**DATA MERGING**:

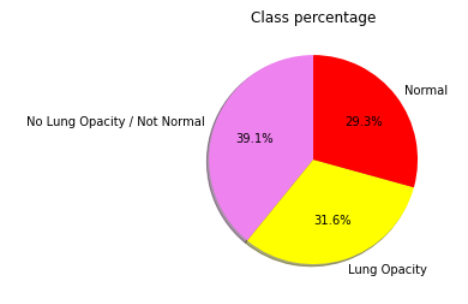


Merged all csv into one data frame with respect to **unique** patient id x1, y1, w1, h1 represents coordinates of first bounding box and x2, y2, w2, h2 represents coordinates of second bounding box.

Note: we are taking only **two** bounding boxes for model building exercise because majority of people are detected by two bounding boxes indicating Lung Opacity. So, our Model will try to predict two boxes for patients with Lung Opacity.

**CLASS AND TARGET INFORMATION**:

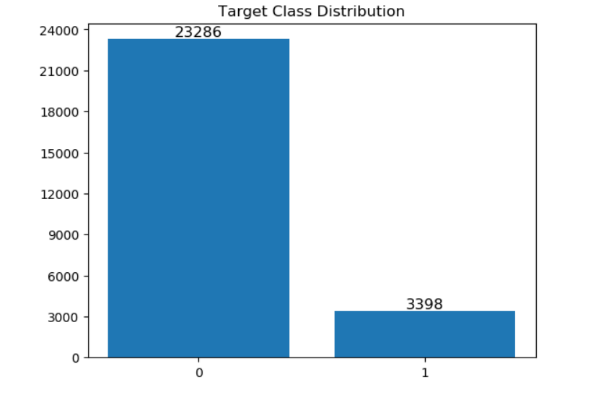
Below visualization represents percentage of class distribution:



**IMBALANCE CLASS PROBLEM**:

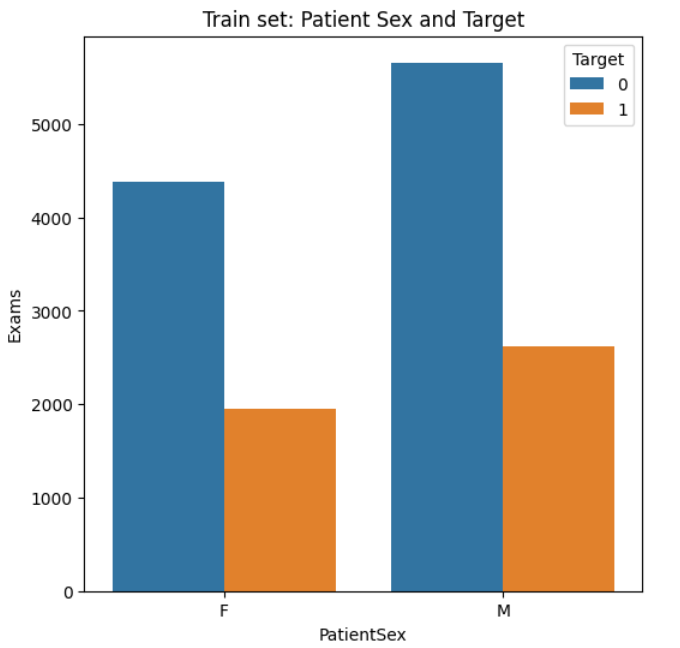
Below bar plot represents number of patients with class label 0 and 1.

With this visualization it is clear that data distribution is not balanced so we need to encounter this issue.

****

**Basic Exploratory Data Analysis on Dataset**:

* Distribution of Target among Males and Females
* Males cases are more as compared to female, but this is not hypothetically true that males are more susceptible to disease.



**EDA ON DICOM IMAGES DATASET:**

Sample of Image of a random patient:



**Dicom images with Bounding box locating lung opacity:**

Below Visualization is showing if the personal has Lung Opacity present then infected area will be highlighted with at least two rectangles.

**A picture containing photo, different, man, room

Description automatically generated**

DATA PRE-PROCESSING

DATA Pre-processing is first step for any AI ML problem and this step is very important for the prediction power of any model.

Here we have images in dicom format so first we need to save them in an array so that we could train model with that and secondly we got images in HD resolution which is **1024x1024** so we need to downsize to **128x128**,reason for using 128 size images is that we will be using Transfer Learning model called **MobileNet** pretrained on ImageNet Dataset.

Since this problem is not only about prediction of classes (classification problem) also we need to predict coordinates of bounding boxes. So, we will be using UNET architecture for the problem solution. For this we created a user defined function and that function return training images array(X\_train) with corresponding mask and target variable.

A screenshot of a cell phone

Description automatically generated

Since we also observed that data distribution is imbalanced for this, we wrote a script which first separate out two classes images into separate variable then combine full set of images with target label 1 and a subset of randomly picked images with label 0(that subset must be of length equal to the length of the full set for target 1)

For instance, if we have 10 images with target label 1 and 100 images with target label 0 then below script will take all 10 images with label 1 and 10 randomly chosen images without replacement from the set of 100 images with label 0 and combine them to create a balanced set of 20 images.

So this way we are able to get 1:1 distribution of classes.

A screenshot of a social media post

Description automatically generated

For UNET we need to feed images as arrays against the mask. So for the prediction of the bounding boxes we first try to predict mask of an image then after we use image processing python package OpenCV to get bounding rectangles and its coordinates. Below is the snapshot of the actual mask of an image.

A picture containing drawing, clock

Description automatically generated

MODEL BUILDING AND EVALUATION

**Training and Testing data splits:**

Before training the model, we need to divide whole set into training and evaluation set and here we for **Classification** model we took first 5000 images as training set and remaining images is for evaluation of the performance on unseen data.

And for the prediction of bounding boxes using **UNET** architecture we are using different set for training and evaluation. As we thought that if the person is normal then we don’t need to do any prediction of boxes for those images. Hence, we are feeding only those images which has some abnormality present and for them we need to highlight those affected area via bounding boxes.

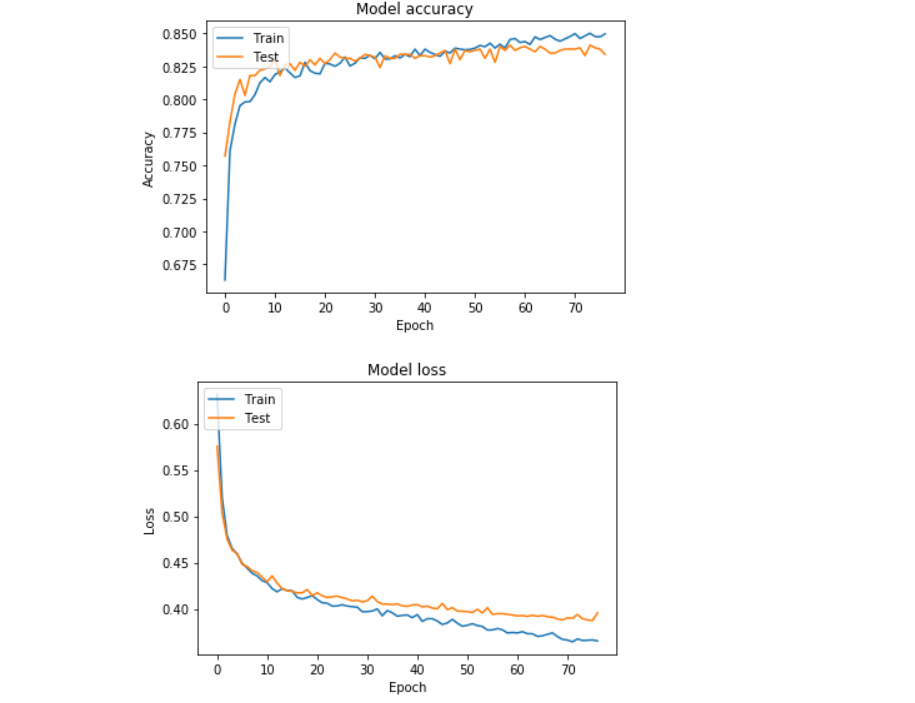
So, for UNET we created another set containing only images with target as 1 (Lung Opacity) and tried to predict mask and bounding boxes thereafter. For Evaluation set we took first 10 images and remaining images for training the UNET model.

**Convolutional Neural Network (Base Model):**

HIGHLIGHTS OF BASE CLASSIFIER MODEL:

* Used TensorFlow 2.0 and Keras wrapper for building classification model.
* Used Conv layers of 64 and 128 features with same padding.
* Applied 2x2 kernel Maxpooling2D layers to make the model lighter.
* Relu activation is used as activation function.
* After flattening of feature vector, we applied 2 dense layers and last layer with SoftMax activation.
* Achieved Evaluation accuracy obtained is 82%

Model Accuracy in basic CNN:



**TRANSFER LEARNING (Model 2):**

We used Transfer Learning model with two branches one for classification other for bounding box prediction using UNet. UNet architecture contains encoder and decoder layers, encoder layers of the transfer learning model are derived from MobileNet architecture and decoder is done using Upsampling of the blocks from the MobileNet architecture until we reach the actual size of the input i.e. 128x128

HIGHLIGHTS OF TRANSFER LEARNING MODEL:

* Used MobileNet pre-trained on ImageNet Dataset
* Encoded the features using MobileNet by removing the tail of dense layers.
* UNet architecture is considered in the project as it is most preferable for MEDICAL IMAGES.
* For computation restrictions as we discussed above that for bounding boxes prediction, we are taking only images with target label 1 (Lung Opacity)
* We are feeding 128x128x3 dimension images into the model and predicting the mask of size 128x128 and similarly we are using the same MobileNet for classification prediction.
* For **Classification** we got 80% of accuracy for evaluation dataset. Model learned fast and moved towards overfit zone but with callbacks we chose decent model for further predictions of test dataset.

**Model Accuracy Visualization and Classification Report**:

A close up of a map

Description automatically generated

A picture containing clock, drawing

Description automatically generated

* For **Mask Prediction** we gotvery low prediction of 60% which is clearly underfit model but due to computations we were not able to make model more deeper (we are using only one conv layer of 16 features after Upsampling) and also we could not choose batch size more than 2 on GPU Machines and for gradient decent batch size plays important role.

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

TEST PREDICTION AND FINAL SUBMISSION EXCEL SHEET

First, we loaded both Classification and Mask Predictor models weights from hard drive location. Note we are using transfer learning model for final classification predictions reason for that is MobileNet took less computation time and will give quicker prediction in production as compared to base CNN model for classification.

For prediction on test dataset images which are present in hard drive location we created a user defined function called **test\_prediction,** the functions takes few random pictures from the path of images and apply all pre-processing steps which we did for images on which we trained the model and then we predicted if the patient has Lung Opacity or not.

Below we discussed the steps if patient has Lung Opacity or No Lung Opacity (Normal):

* If dicom image of a patient predicted the person has Lung Opacity, then the same image will go into our second model which is UNet model for prediction of mask. After prediction of mask we used Image processing library OpenCV to find **Contours** and each contour predicted 4 coordinates of bounding boxes.

Since our model was underfit and not giving accurate mask prediction, from any predicted mark we were getting more than 2 contours hence more than two bounding boxes for each image. To solving this issue, we first used Non-Max Suppression (NMS) but still we were getting noisy small bounding boxes along with two larger boxes. Then after we used a simple **mathematics** approach, we calculated area of rectangle for each predicted box and chose top two boxes based on maximum area.

* If dicom image of a patient predicted the person is normal and no anomaly is present in his/her X-ray, then in this case we don’t need to call UNet Model for prediction of any bounding boxes. So, in this case we updated all coordinates as zero.

**Predicting Result:**

Below image is highlighting anomaly area in both 128x128 resolution(left) and 1024x1024 resolution(right):

A picture containing photo, indoor, looking, different

Description automatically generated

**Submission:**

For submission we created similar function as above but only changes we made that this function returns only data frame and we exported the data like sample submission.

**Learnings and Improvements**

With this Capstone Project we learned how to implement Object Detection on medical Images and we faced few challenges mainly related to computers and powerful GPU Machines as Google Colab is good for small Deep Learning Problems but for problem like this we need Big Nvidia GPUs, not single GPU we need mutli GPUs for parallel processing.

For Improvements we may fit this model on big GPU machine to get appropriate results as we tested few epochs with deep CNN UNet and 32 batch size, model was doing better than the above UNet but each epoch was taking more than 2 hours to process. So, we dropped the idea of training such a complex model.

GitHub link:

Team Members Name: Dhruv Khanna, Harish, Saurav, Kranti and Ragu.

Submission Date: 17/05/2020

**We declared hereby that every line of code is handwritten by team members and not copied from any resources available online.**