

**PGP AI ML – Capstone Final Submission**

**Pneumonia Detection Challenge**

**Group A CV Batch 1**

**Table of Contents**

[Summary of problem statement, data and findings 3](#_Toc87791104)

[Overview of the final process- data pre-processing steps, the algorithms used 3](#_Toc87791105)

[Step-by-step walk through the solution 4](#_Toc87791106)

[Model evaluation- What parameters were prominent and how did we evaluate the success of the models 8](#_Toc87791107)

[How does the final solution compare to the benchmark laid out at the outset? 9](#_Toc87791108)

[Visualizations – Exploratory Data Analysis 10](#_Toc87791109)

[Model Deployment 14](#_Toc87791110)

[Implications 14](#_Toc87791111)

[Limitations 16](#_Toc87791112)

[Closing Reflections 16](#_Toc87791113)

### Summary of problem statement, data and findings

The Real problem - Pneumonia accounts for over 15% of all deaths of children under 5 years old internationally. Accurately diagnosing pneumonia 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. A number of factors such as positioning of the patient and depth of inspiration can alter the appearance of the CXR, complicating interpretation further. 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

We need to build a Pneumonia detection model to detect a visual signal for pneumonia in medical images. Specifically, the algorithm needs to automatically locate lung opacities on chest radiographs, providing affected area details through bounding box

Data - Data provided is Dicom original images: - Medical images are stored in a special format called DICOM files (\*.dcm). They contain a combination of header metadata as well as underlying raw image arrays for pixel data. While we are theoretically detecting “lung opacities”, there are lung opacities 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. Lung Opacity class refers to Pneumonia cases. There are around 26000 2D single channel CT images in the pneumonia dataset that provided in DICOM format.

Findings-

* “Faster R-CNN” model predicting bounding boxes of pneumonia patients with confidence intervals.
* VGG19 seems to be the best model with highest recall, we will continue with Faster RCNN since this is an object detection problem

### Overview of the final process- data pre-processing steps, the algorithms used

Like most data science undertakings, we followed the CRISP-DM approach for this project. Since the dataset comprised of Dicom images, we installed Pydicom library to read the images. For the detectron2 model, we also used Albumentations library for Data augmentations on the data set such as Horizontal flip, Cropping, Scaling and so on.

We performed a number of diagnostics on the data using visualizations to gather preliminary insights. Next up, we ran a bunch of models such as VGG19, Resnet 50, InceptionNet v3, U-Net and Faster R-CNN.

We finally evaluate all model performance using classification Matrix and Mean Average Precision (mAP) to select and pickle our model that can be used at the backend of API for further predictions.

Below is a figure depicting the process flow of our project approach -

Diagram

Description automatically generated with medium confidence

Figure 1: Project Approach

### Step-by-step walk through the solution

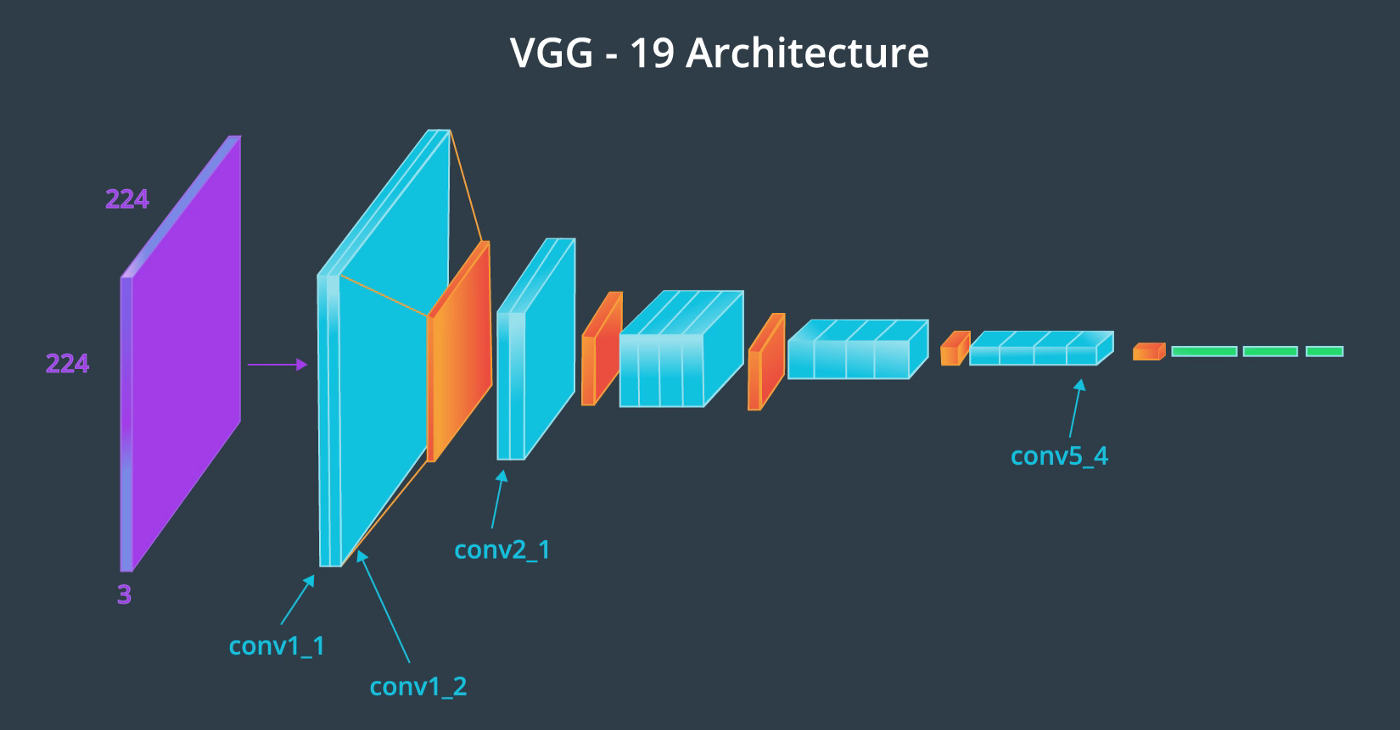
As mentioned previously, we deployed a number of models such as VGG19, ResNet 50, InceptionNet v3, U-Net and Faster R-CNN

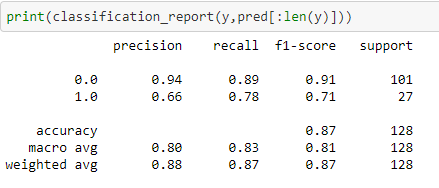
#### VGG19

VGG19 is a variant of VGG model which in short consists of 19 layers (16 convolution layers, 3 Fully connected layer, 5 MaxPool layers and 1 SoftMax layer). There are other variants of VGG like VGG11, VGG16 and others. VGG19 has 19.6 billion FLOPs.

The input to VGG based convNet is a 224\*224 RGB image. Preprocessing layer takes the RGB image with pixel values in the range of 0–255 and subtracts the mean image values

which is calculated over the entire ImageNet training set.VGG19 Variation has 19 weight layers consisting of 16 convolutional layers with 3 fully connected layers and same 5 pooling layers.



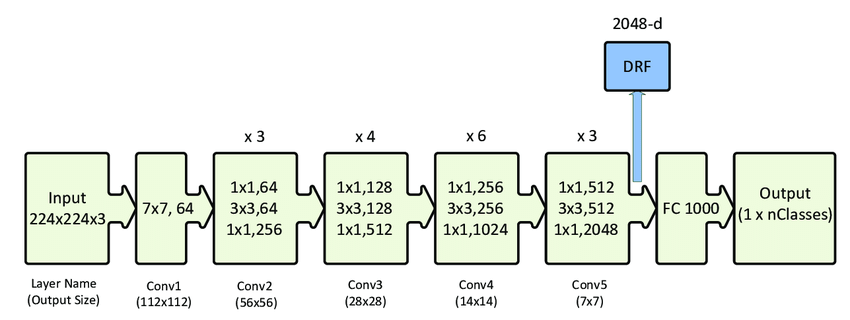


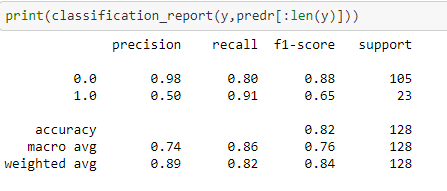
#### ResNet 50

ResNet-50 is a convolutional neural network that is 50 layers deep. Note that there is only one 3x3 convolution rather than two. 1x1 convolutions are used to map in lower dimension and then perform 3x3 convolution and then remap them to higher dimensions. This way the training time will be less.

The other part to note is in ResNet 50 when there is dimension change then the authors used 1x1 convolutions at x to make the dimension same.

#### ResNet 50 architecture





#### Inception Net v3

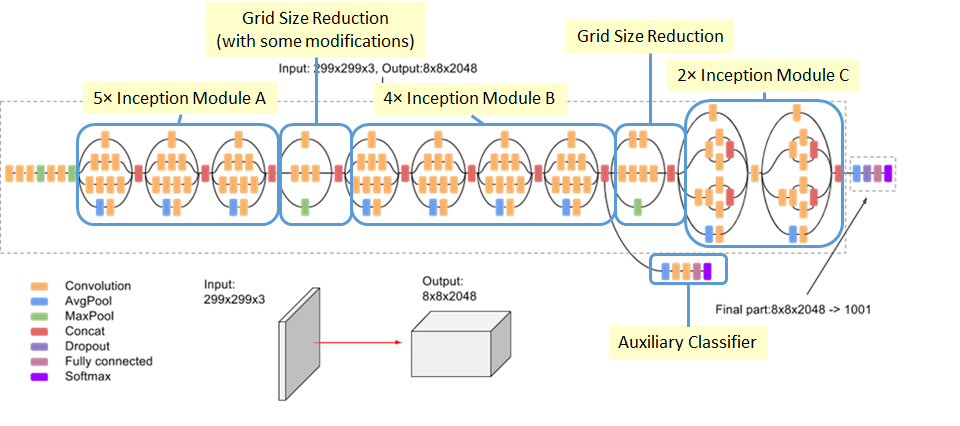
**Inception-v3** is a convolutional neural network architecture from the Inception family that makes several improvements including using [Label Smoothing](https://paperswithcode.com/method/label-smoothing), Factorized 7 x 7 convolutions, and the use of an auxiliary classifer to propagate label information lower down the network (along with the use of [batch normalization](https://paperswithcode.com/method/batch-normalization) for layers in the sidehead).

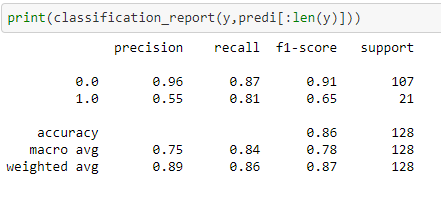
The architecture of an Inception v3 network is progressively built, step-by-step, as explained below:

1. Factorized Convolutions: this helps to reduce the computational efficiency as it reduces the number of parameters involved in a network. It also keeps a check on the network efficiency.

2. Smaller convolutions: replacing bigger convolutions with smaller convolutions definitely leads to faster training. Say a 5 × 5 filter has 25 parameters; two 3 × 3 filters replacing a 5 × 5 convolution has only 18 (3\*3 + 3\*3) parameters instead.

#### Inception V3 Architecture

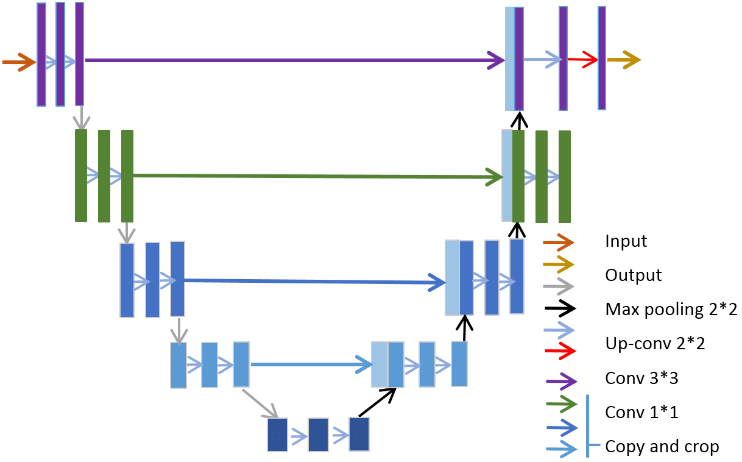




#### U-Net

U-Net architecture which is used for image segmentation and Works efficiently on small datasets through heavy data augmentation as in some cases the number of annotated samples will be less. The features extracted at different levels in the contracting path are then combined with the feature maps in the expansion path giving rise to more number of features which is necessary for an accurate segmentation map.

#### U-Net Architecture



BASIC UNET MODEL : CLASSIFICATION REPORT

precision recall f1-score support

0 0.92 0.62 0.74 2068

1 0.38 0.81 0.52 601

accuracy 0.66 2669

macro avg 0.65 0.71 0.63 2669

weighted avg 0.80 0.66 0.69 2669

Graphical user interface

Description automatically generated

#### Faster R-CNN using Detectron2

Faster R-CNN model was developed using Detectron2 on

Pytorch. All dicom images are converted to JPG as

Detectron2 can't directly read dicom images. We prepare

the dataset into Json format and then feed into

the Detectron2 model for training.

#### Augmentation:

Standard augmentation function for images with bounding

boxes was not feasible and hence we developed a

pipeline using Albumentations library from PyPi.

Augmentation was performed only on 10% of

dataset sampled randomly.

The X-ray picture shows that augmentation is working with

**horizontal flip of image** with bounding box.

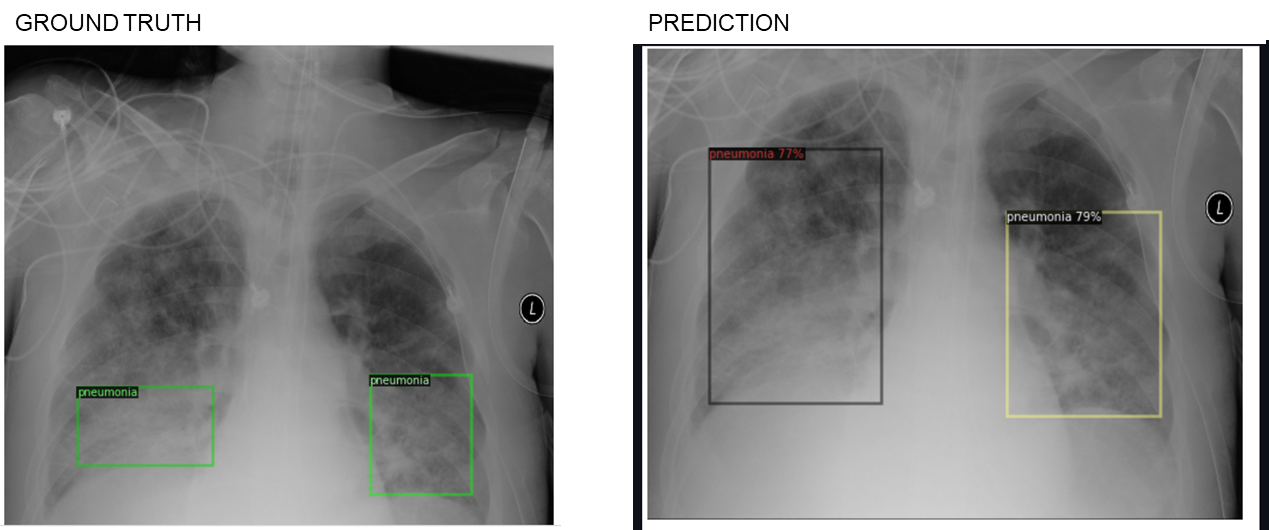
Table

Description automatically generated

We apply transfer learning of the Faster R-CNN model i.e. pre-trained model on the COCO dataset. We split the data into 90:10, trained the model. The code is developed using RoIAlign instead of RoIPooling

Detectron2 is built using Pytorch, and provided a very easy API. Detectron2 originates from Mask R-CNN benchmark. Detectron2 helps in Panoptic segmentation ( combination of semantic and instance based). It can also be used as a wrapper on top of other projects and can be exported to easily accessible formats. Detectron2 is flexible and fast training on single or multiple GPU servers.

The Faster R-CNN model is trained to predict the bounding box of the pneumonia area with a confidence score as shown below against the ground truth. We used Visualizer utility to draw the predictions on the image



Since the problem statement requires us to read images, we use CNN and various state of the art deep learning architectures listed here to identify lung opacity with higher recall and accuracy. We have run these architectures on full data set except UNet which was processed on subset of data

### Model evaluation- What parameters were prominentand how did we evaluate the success of the models

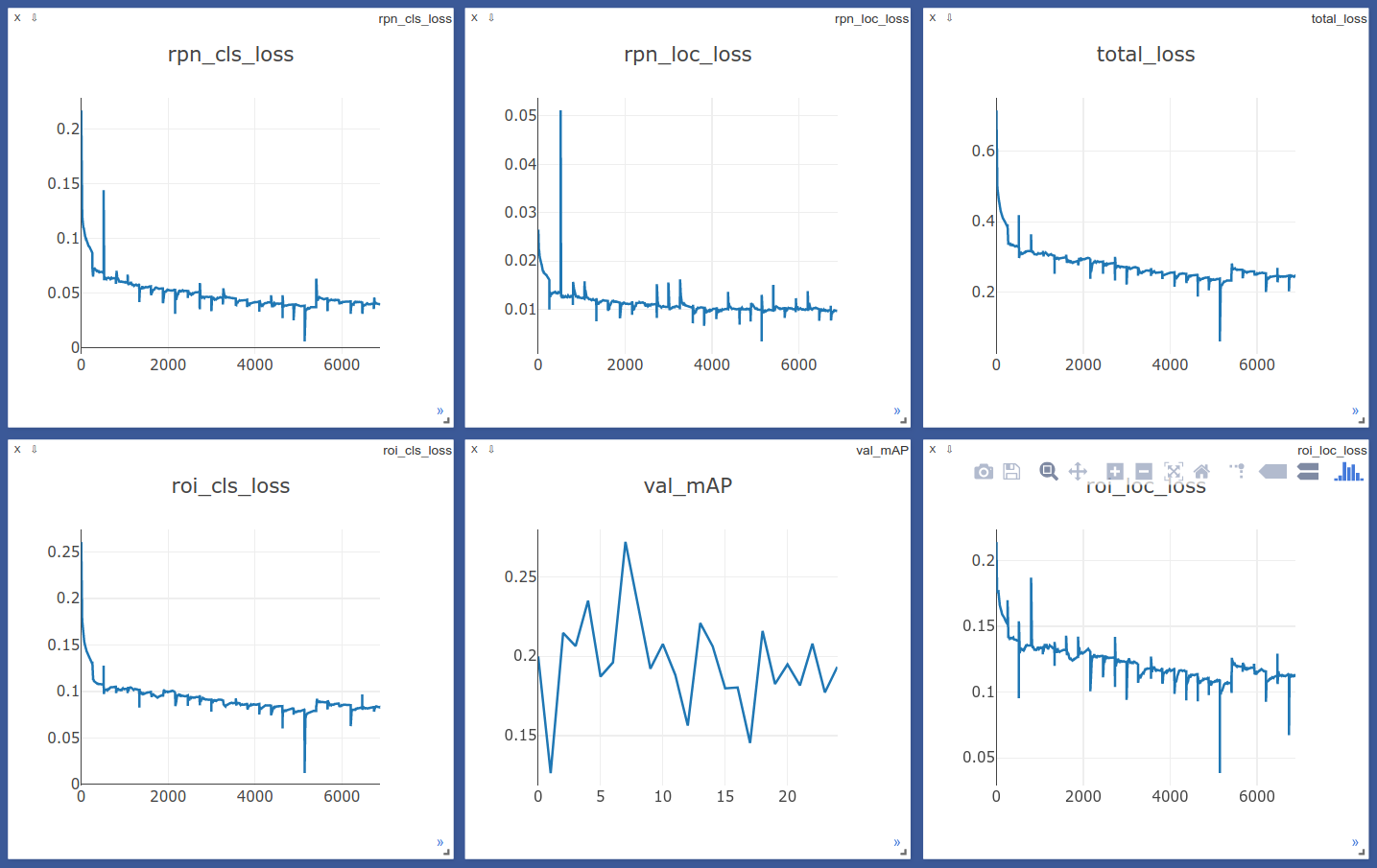
Since the problem statement requires us to read images, we use CNN and various state of the art deep learning architectures listed here to identify lung opacity with higher recall and accuracy. We have run these architectures on full data set except UNet which was processed on subset of data

Table

Description automatically generated

Although from the consolidated results, **VGG19** seems to be the best model with highest recall, we will continue with **Faster R-CNN** since this is an object detection problem

We use classification matrix to check for precision values of each model. Since it’s an object detection problem, we finalize faster R-CNN among all the architectures. This project has also been evaluated on the mean average precision at different intersection over union (IoU) thresholds for a set of predicted bounding boxes and ground truth bounding boxes. The metric sweeps over a range of IoU thresholds, at each point calculating an average precision value. A true positive is counted when a single predicted object matches a ground truth object with an IoU above the threshold. A false positive indicates a predicted object had no associated ground truth object. A false negative indicates a ground truth object had no associated predicted object.



### How does the final solution compare to the benchmark laid out at the outset?

We expected high mean average precision of ### at the outset but did not quite achieve the same. We measured it for various IoUthresholds

### Visualizations – Exploratory Data Analysis

From our EDA, we learned that there are 26684 unique patients. Few patients have multiple bounding boxes -

* 1. 3266 patients have 2 bounding boxes defined
  2. 119 patients have 3 bounding boxes defined
  3. 13 patients have 4 bounding boxes defined

We will run a binary classification to predict patients with pneumonia. Target – 1 indicates patients with pneumonia and target – 0 indicates normal patients or patients having lung opacity but not pneumonia

8,851 (29.3%) patients are healthy/Normal

9,555 (31.6%) patients have Lung Opacity

11,821 (39.1%) patients have No Lung Opacity but may have other lung abnormality

Clearly, dataset is bit imbalanced with Target class being only 31.6% of the whole dataset

Chart, pie chart

Description automatically generated

Chart, pie chart

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Patient distribution shows most pneumonia patients or patients are found between age 40-60. More records for male is observed in the data. Proportion of pneumonia and normal patients re equal for female patients. However, for males, proportion of pneumonia patients are more than normal

Chart, histogram

Description automatically generated

Chart, histogram

Description automatically generated

Chart, histogram

Description automatically generated

Chart, bar chart

Description automatically generated

Chart, bar chart

Description automatically generated

Lung opacity is seen maximum in age 50-65 followed by age group 20-50

Chart, scatter chart

Description automatically generatedChart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generatedChart, scatter chart

Description automatically generated

Chest X-ray can be done in **Posterior-Anterior (PA)** projection and **Anterior-Posterior (AP)** Projection. Lung opacity seems to be more visible in AP position

Chart, bar chart

Description automatically generatedChart, bar chart

Description automatically generated

Chart, bar chart

Description automatically generated

Visualization of sample X-ray images for each category

A collage of a person's face

Description automatically generated with medium confidence

Visualization of sample X-ray images with bounding boxes depicting lung opacity

A picture containing graphical user interface

Description automatically generated

A collage of a person's face

Description automatically generated with medium confidence

### Model Deployment

Model deployment

### Implications

The size, shape, and position of pneumonia can vary a great deal. Its target contour is very vague, which leads to great difficulty with detection, and enhancing the accuracy of detection is a major research problem. Medical testing has high requirements for accuracy, and hence two-stage detectors such as **Faster R-CNN** have an advantage in this respect.

Graphical user interface

Description automatically generated

Graphical user interface, website

Description automatically generated

### Limitations

There are still problems with the backbone network of the current detection algorithms. For example, ResNet generally has two problems: a large network depth leading to long training time and massive down-sampling that leads to the target position and semantic information being lost

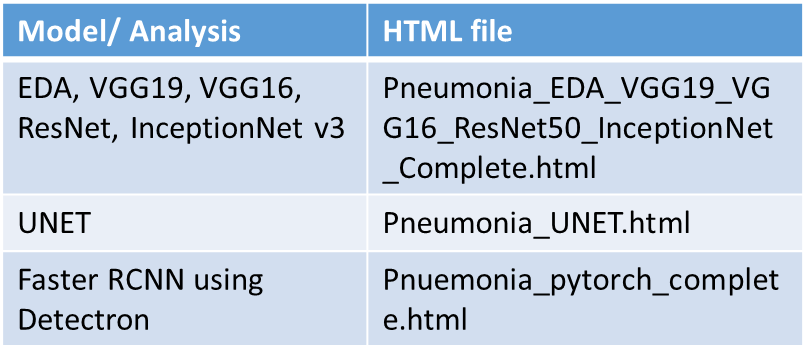
The dataset contains three categories of subjects, normal, pneumonia, and abnormal(cancer or other diseases) but only provides the bounding box for pneumonia images. However, the features of pneumonia and abnormal(cancer or other diseases) are pretty similar, which caused the failure to distinguish pneumonia and abnormal images for Faster R-CNN. This results in predicting bounding box for abnormal images

Different distribution of train and test datasets, most likely due to different labeling methodology

### Closing Reflections

We had class imbalance for pneumonia cases in our dataset. Hence, using NIH dataset since It's a bigger dataset but with lower quality of labels, would be very interesting to check if training the model to predict both datasets would improve the result, or at least use it to pretrain the base model

### Code Base

****

### References