Pneumonia Detection in Chest Radiographs

Dr. Abhishek Kadam| Amit Mahajan| Aparna Nammi| Deepak Upadhyay| Sriram Srinivasan

29/Nov/2020

Table of Contents

Contents

Abstract

Globally, Pneumonia is a one of the top causes of deaths by an infection. The disease is caused by Bacteria, Fungi, and viruses. It is characterized by lung inflammation leading to filling up of lung tissues with fluid or pus or both. A successful diagnosis of Pneumonia is based on the study of the patients, sign and symptoms, medical history of patients and diagnostic tools like X-Ray chest (CXR). Our objective is to build an algorithm to detect a visual signal for pneumonia in medical images. Specifically, our algorithm automatically locates lung opacities, a hallmark of pneumonia, in chest radiographs. We have used R-CNN model for building this detector. In this report we describe our novel approach to solve this problem for Indian clinical practice and environment.

Introduction

About Pneumonia

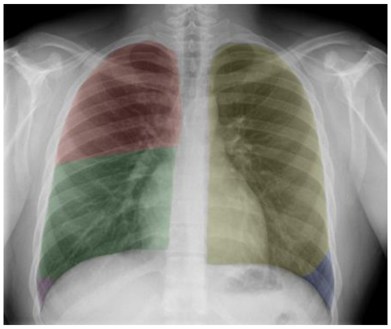
A 2019 article on stoppneumonia.org (stoppneumonia, 2019) indicates Pneumonia is the world's leading infectious killer of children, claiming the lives of more than 800,000 children under the age of five every year, more than 2,000 every day. One of the global calls for action on the childhood Pneumonia is to prioritize research, development, and innovation to improve access to the most affordable and cost-effective pneumonia prevention, diagnosis, referral and treatment technologies and services (stoppneumonia, 2019). In 2015, in India only 78% children with pneumonia symptoms were taken to the health facility. 44.5 per 10,000 people is the minimum number of skilled health workers required to deliver quality health services as per WHO recommendations. In India we had 8 doctor per 10,000 people and 10 nurses/midwives per 10,000 people. The diagnosis of Pneumonia is therefore dependents on these healthcare workers. It is evident that in India we have only 1/3rd of the minimum skilled health workers to deliver quality of health services. It is imperative that there is innovation in the healthcare sector that reduces the burden on the health care system. One way of doing so is to find a cost effective and automated way of reading the CXRs which is one of the primary diagnostic tools for Pneumonia.

Pneumonia Radiography

Before getting into the Pneumonia radiography, it is important to understand the normal CXR shows contours and shadows of organs such as heart, diaphragm, ribs etc.

Normal CXR

Figure 1



Case courtesy of Dr Phillip Marsh, Radiopaedia.org, rID: 58938

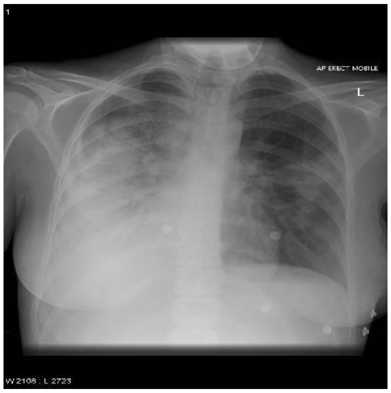
1. Red: Right upper lobe (RUL)
2. Green: Right middle lobe (RML)
3. Purple: Right lower lobe (RLL)
4. Yellow: Left upper lobe (LUL)
5. Blue: Left lower lobe (LLL)

A skilled radiographer can navigate through the normality and abnormalities in the CXR before confirming the diagnosis on Pneumonia.

A typical CXR of a patient showing Pneumonia signs and symptoms is characterized by opacities. These opacities may be confluent or patchy depending upon the amount and distribution of the infection in the lungs. The CXR may show reticulo-nodular pattern, i.e. an overlap of net-like (reticular) shadows and nodular shadows, indicative of interstitial Pneumonia.

Pneumonia CXR

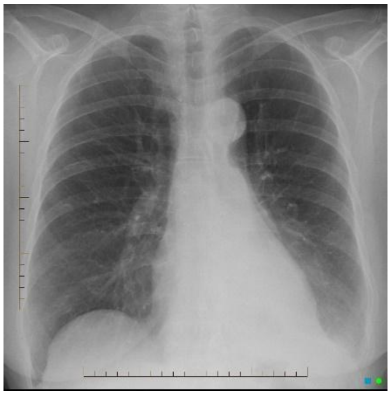
Figure 2



Case courtesy of Assoc Prof Frank Gaillard, Radiopaedia.org, rID: 11009

Some other lung diseases may also show opacities in the CXR and must be carefully differentiated from Pneumonia. In Atelectasis – or collapse of lungs is associated with the increased density of the lung tissue. However, it is easily differentiated with Pneumonia due to the loss in the volume of lung/s in Atelectasis.

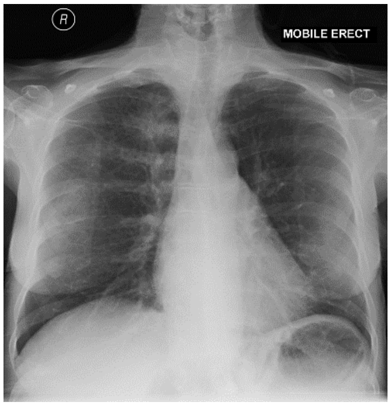
Figure 3



Case courtesy of Assoc Prof Frank Gaillard, Radiopaedia.org, rID: 7312

Adenocarcinoma for Lungs typically affects the entire lobe of the lung.

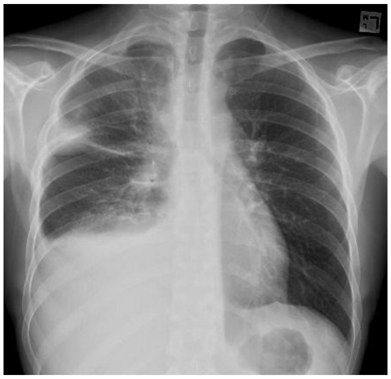
Figure 4



Case courtesy of Dr Henry Knipe, Radiopaedia.org, rID: 44103

Plumonary Lymphoma also can affect the entire lobe of lung.

Figure 5



Case courtesy of Dr Jack Ren, Radiopaedia.org, rID: 29094

Computer Vision Algorithms for object detection challenge.

Computer vision is a vast arear of applied deep learning which has in recent years not only provided near human-level performance, but also bettered it in several areas <Insert examples>. The Convolutional Neural Networks (CNN) have opened doors for classification, segmentation, and object detection in the images. The availability of GPUs to train these models and development of concept like transfer learning have now increased the favorability to the deep learning models for some overly complex tasks. It is not a surprise that many medical imaging devices now assisting the clinicians to detect abnormalities in the images and flag such case. Even today, the CNN algorithms are assisting doctors for anomaly detection. With the availability of Radiological Society of North America (RSNA®) dataset since 2018, the efforts have now moved from anomaly detection to anomaly localization. RSNA has collaborated with US National Institutes of Health, The Society of Thoracic Radiology, and MD.ai to develop a rich dataset. The dataset consists of close to 30000 CXR images of lungs. The data also consist of 3 .csv files each providing details that have been put together by professionals / experts. Since the availability of this well curated data sets, there has been a huge interest in developing bounding box models to locate the anomaly – in this case the opacity in CXR. This is a challenge where the CXR must be processed to detect inflammation of the lungs by developing an algorithm to detect a visual signal for Pneumonia in the CXR. The algorithm needs to automatically locate lung opacities in CXRs.

This is a classification + localization problem. Typically, we use the CNN with Softmax for classification and output the location of the object/s. The location of the object/s in specified by identifying the coordinates of center of the object and then determining the height and width of the object/s. Each object with have four outputs that will tell the location of the class in the image: in this case the CXR and the opacity in the CXR.

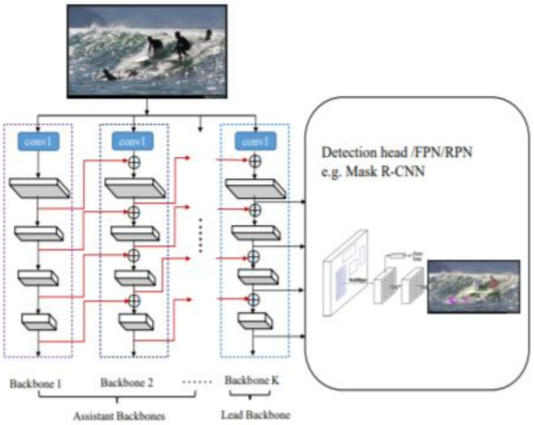
While there were brute force approaches in the past, currently the Single Shot Detection (SSD) and You only Look Once (YOLO) are state of the art one step models that are predominantly used for object detection. These methods give priority to the inference speed.

There are other set of state-of-the-art models such as Faster- RCNN, Mask RCNN, Cascade RCNN which are also effective and give priority to the accuracy of detection of object. These are region proposal based algorithms

CB Net (Yudong Liu, 2019) – A novel approach

All the above approaches use a backbone network, and the performance of these object detectors is highly dependent on the feature extraction done by the backbone network. Composite Backbone Network (CB Net) (Yudong Liu, 2019) uses the strategy of assembling multiple identical backbones by composite connections between the adjacent backbones, to form a more powerful backbone. Each backbone provides the high level feature map and input features to the succeeding backbone and finally the feature maps of lead backbone (the last backbone) are used for object detection. The method seems to increase the mAP for the FPN, Mask RCNN and Cascade RCNN on the benchmark COCO dataset (Yudong Liu, 2019). See Figure 6 below; the red arrows indicate composite connections.

Figure 6 Illustration of CB NET



While the above approach is using Adjacent Higher-Level Composition (AHLC), the same paper proposes alternative composite approaches such as Adjacent Lower-Level Composition (ALLC), Same Level Composition (SLC) and Dense Higher-Level Composition (DHLC) in case of DenseNet.

DeTraC (Abbas, 2020) – A Decompose, Transfer and Compose approach

In this approach the authors explain how the DeTraC approach can deal with any irregularities in the image dataset by investigating its class boundaries using a class decomposition mechanism. They developed this model to detect COVID-19 CXR images from normal and Severe Acute Respiratory Syndrome

(SARS) with an accuracy of 93.1% and sensitivity of 100%.

DeTraC establishes the effectiveness of class decomposition in detecting COVID-19 from CXR images. The main contribution of DeTraC is its ability to deal with data irregularities, which is one of the most challenging problems in the detection of COVID-19 cases. The class decomposition layer of DeTraC can simplify the local structure of a dataset with a class imbalance (Abbas, 2020). The DeTraC model is an approach for classification of the COVID-19 CXRs however it may not be used for localization.

Figure 7 Decompose, Transfer, and Compose (DeTraC) model for detection of COVID-19 from CXR images

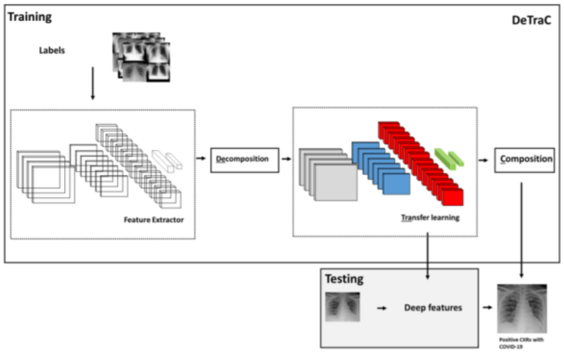
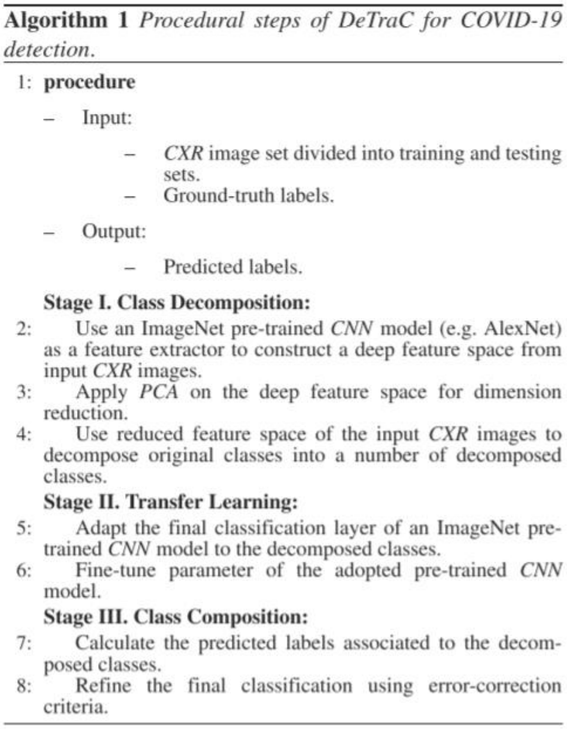


Figure 8 Procedural steps of DeTraC for COVID-19 detection



Algorithm 1 *Procedural steps of DeTraC for COVID-19 detection.*

```
procedure
    – Input:
        – CXR image set divided into training and testing sets.
        – Ground-truth labels.
    – Output:
        – Predicted labels.
    Stage I. Class Decomposition:
    Use an ImageNet pre-trained CNN model (e.g. AlexNet) as a feature extractor to construct a deep feature space from input CXR images.
    Apply PCA on the deep feature space for dimension reduction.
    Use reduced feature space of the input CXR images to decompose original classes into a number of decomposed classes.
    Stage II. Transfer Learning:
    Adapt the final classification layer of an ImageNet pre-trained CNN model to the decomposed classes.
    Fine-tune parameter of the adopted pre-trained CNN model.
    Stage III. Class Composition:
    Calculate the predicted labels associated to the decomposed classes.
    Refine the final classification using error-correction criteria.
```

Data and Insights

Data

Source

The goal of this project is to develop a model that can detect the opacities in a CXR. For the exploratory data analysis and subsequent model building we will assume that the presence of opacities indicates presence of Pneumonia. The data is downloaded from Kaggle.

Datasets

We have used

- the images for the current stage i.e. stage 2 – provided as stage _2_train_images.zip.
- the training data - stage_2_train_labels.csv.
- the sample submission stage_2_sample _submission.csv, which provides the IDs for the test set.
- the file stage_2_detailed_class_info.csv contains detailed information about the positive and negative classes in the training set and may be used to build more nuanced models.

Data Format

The training data is provided as a set of patientIds and bounding boxes. Bounding boxes are defined as follows: x-min y-min width height

There is also a *binary* target column, Target, indicating pneumonia or non-pneumonia.

There may be multiple rows per patientId.

DICOM Images; All provided images are in DICOM format.

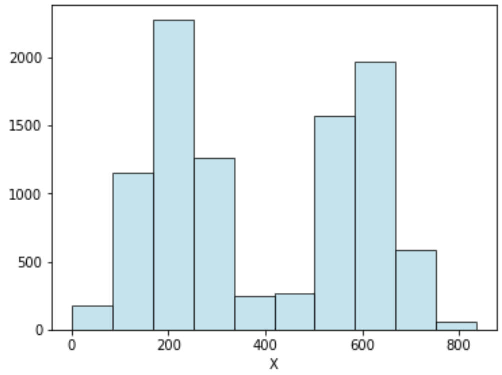
Exploratory Data Analysis (EDA)

File 1 - stage_2_train_labels.csv

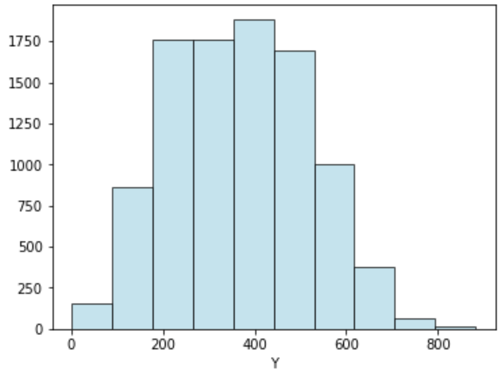
This file has six columns.

“patientId” has 26684 unique values.

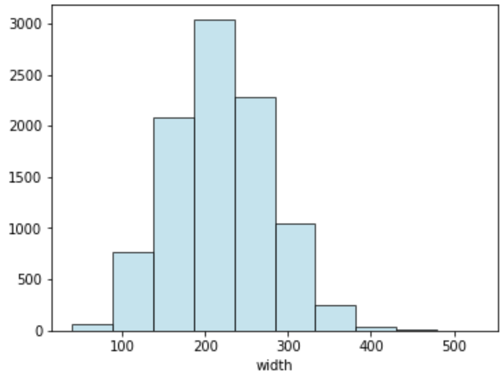
“x” has 2 distinct groupings of data. The first group has a range of values from 2.0 to 418.50. The second group has a range of values from 418.50 to 835.0



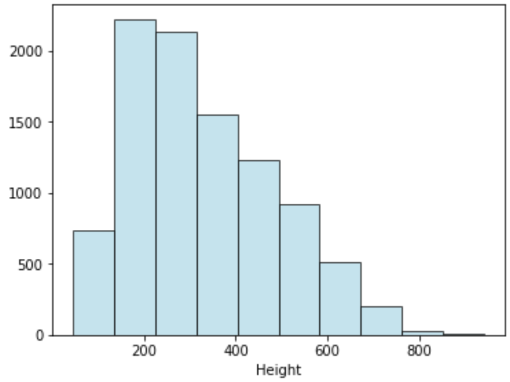
“y” has a normal distribution.



“width” has a median of 217.0 pixels and the minimum and maximum width of the bounding box is 40.0 and 528.0, respectively.



“height” has a median of 298 pixels and the minimum and maximum height of the bounding box is 45.0 and 942.0, respectively.



“Target” has 2 value: “0” = non-pneumonia and “1” = pneumonia. There are 20672 values between 0.00 and 0.05 and 9555 values between 0.95 and 1.00. The data is skewed towards the Target class non-pneumonia. This skewness needs to be addressed in the dataset to avoid a potential bias towards non-pneumonia class in training the model. This is particularly important as our model must be designed to have minimal false negatives i.e. classifying “Non-Pneumonia” when it is actually a “Pneumonia”.

We will see further how the imbalance was treated to achieve a normalization of data.

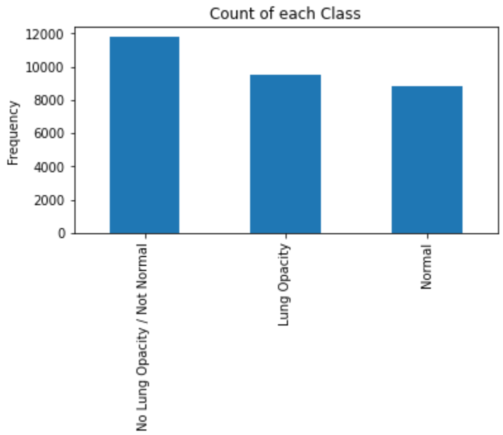
File 2 - stage_2_train_labels.csv

This file is 2 columns.

“patientId” has 26684 unique values

“class” has 3 distinct classes.

1. No Lung Opacity / Not Normal (39%)
2. Lung Opacity (32%)
3. Other (29%)



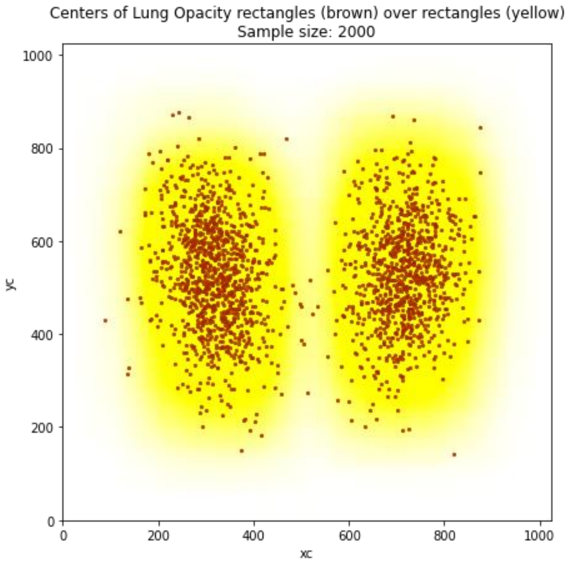
The above percentages further confirm that the class imbalance exists and needs to be treated.

It is evident that the CXR for one patient may have more than one bounding box. Maximum number of bounding boxes for a single CXR is observed to be four.

Number of bounding boxes	Number of patientIds
1	23286
2	3266
3	119
4	13

We merged the File 1 and File 2 to get a homogenous dataset for further EDA.

Our EDA of 2000 patientIds revealed that the Centers of Lung Opacity rectangles showed high concentration near the central portion of the lungs.

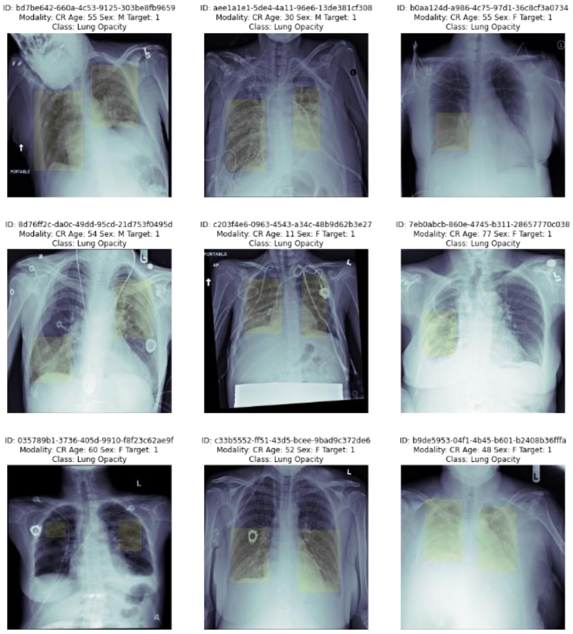


File 3 – stage_2_train_images

Exploration on the DICOM images led to understand the metadata of the DICOM images. Of the several metadata items we performed an EDA using following metadata items –

1. Patient sex. – The CXRs belong to both male and female patients.
2. View position - Both AP and PA body positions are present in the data. The meaning of these view positions is [2] [3]: AP - Anterior/Posterior; PA - Posterior/Anterior.
3. Rows & Columns – These indicate the size of each DICOM image. The DICOM images in this dataset are 1024 X 1024. The size of the file must be downscaled for the model training.

Illustration of the DICOM image overlayed with the bounding box from the File 1.


ID: bd7be642-660a-4c53-9125-303be8fb9659
Modality: CR Age: 55 Sex: M Target: 1
Class: Lung Opacity
PORTABLE
ID: aee1a1e1-5de4-4a11-96e6-13de381cf308
Modality: CR Age: 30 Sex: M Target: 1
Class: Lung Opacity
ID: b0aa124d-a986-4c75-97d1-36c8cf3a0734
Modality: CR Age: 55 Sex: F Target: 1
Class: Lung Opacity
ID: 8d76ff2c-da0c-49dd-95cd-21d753f0495d
Modality: CR Age: 54 Sex: M Target: 1
Class: Lung Opacity
ID: c203f4e6-0963-4543-a34c-48b9d62b3e27
Modality: CR Age: 11 Sex: F Target: 1
Class: Lung Opacity
PORTABLE
AP
ID: 7eb0abcb-860e-4745-b311-28657770c038
Modality: CR Age: 77 Sex: F Target: 1
Class: Lung Opacity
ID: 035789b1-3736-405d-9910-f8f23c62ae9f
Modality: CR Age: 60 Sex: F Target: 1
Class: Lung Opacity
L
ID: c33b5552-ff51-43d5-bcee-9bad9c372de6
Modality: CR Age: 52 Sex: F Target: 1
Class: Lung Opacity
L
ID: b9de5953-04f1-4b45-b601-b2408b36fffa
Modality: CR Age: 48 Sex: F Target: 1
Class: Lung Opacity
L

Treating the data imbalance

The data imbalance is due to the need of classifying the CXRs as Pneumonia v/s Non-Pneumonia. Approximately 2/3rd of the training data labels as well as the DICOM images will be Non-Pneumonia.

Training a model on such a data will result in the model being biased toward the Non-Pneumonia CXRs.

Normalizing the data

The data was normalized by randomly selecting 9555 units of Non-Pneumonia class. Post normalization, the data is now balanced to have equal proportion of Pneumonia and Non-Pneumonia classes.

Data Augmentation

The data will be augmented as necessary for the models. For the initial EDA we have not augmented.

Insights

Summary

Some of the key insights from our EDA are summarized below: -

1. There is a binary classification of the CXRs, Pneumonia and Non-Pneumonia.
2. The bounding box co-ordinates are available in the dataset and are represented as x, y, w (width) and h (height)
3. **This is a classification + regression problem** as the model will first need to learn the classify Pneumonia and Non-Pneumonia from the CXR and followed by predicting the position of the bounding box which means predicting the location of the opacity i.e. the pneumonia patch.

Forward path

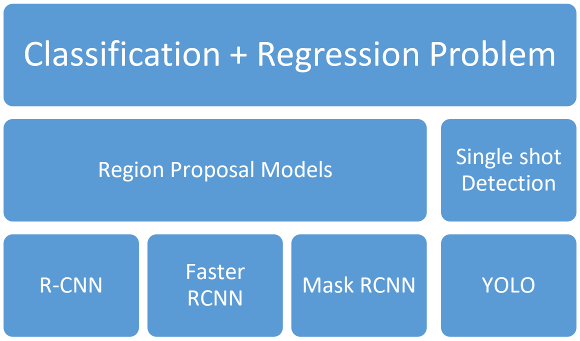
As we have established that the Pneumonia detection is a classification + regression problem; we will be exploring the object detection models for creating a solution of this problem.

We also understand that the performance of this model is best assessed by using Intersection over Union (IoU), using mean average precision (mAP) as a metric.

Our Approach

Objective

The overall objective is to build a model that will be able to *classify* the CXRs as Pneumonia (Lung Opacity) or Non-Pneumonia (No Lung Opacity) and then *predict* the bounding box/s position on the CXR. For this our objective is to build a baseline model based on CNN and a Sigmoid function. We expect to get a model accuracy of approximately 80 percent to start with. We plan to use state of the art models as well.



Hypothesis/Tests

The model we will build is for the detection on Pneumonia in the CXRs. As explained for an objective such as detection on existence of a disease from the image, it is particularly important that **the model predicts with high recall** i.e. has minimal false negatives.

The model shall be built on a normalized data such that there is a 50-50 representation of the two classes i.e. Pneumonia and Non-Pneumonia.

We plan to assign a proper loss function that is sensitive to changes and does not round off the values unnecessarily. (Arjun_Kashyap, 2019) The main key is the threshold value that you assign to your final layer of the neural network. Since this is a case of binary classification, we must set the threshold in such a way as to maximize recall. We will try to maximize recall by setting the threshold below 0.5 i.e., around 0.2.

We may have to also balance between Precision and Recall

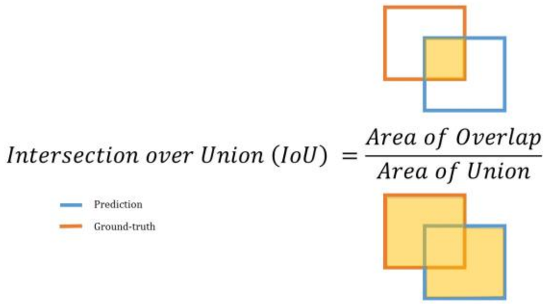
Cost/Loss Function and Metric

Evaluation Metric - Intersection over Union (IoU)

The RSNA Pneumonia detection was evaluated on the mean average precision at different intersection over union (IoU) thresholds. For would project we will be following the same evaluation method.

We choose this cost function because IoU is a measure of the magnitude of overlap between two bounding boxes i.e. the predicted box and the ground truth (or, in the more general case, two objects). It calculates the size of the overlap between two objects, divided by the total area of the two objects combined.

It can be visualized as the following:


Intersection over Union (IoU) = Area of Overlap / Area of Union
Prediction
Ground-truth

In the above example, the two boxes in the visualization overlap, but the area of the overlap is insubstantial compared with the area taken up by both objects together. IoU would be low - and would likely not count as a "hit" at higher IoU thresholds.

The IoU of a set of predicted bounding boxes and ground truth bounding boxes is calculated as:

$$\text{IoU}(A, B) = \frac{A \cap B}{A \cup B}$$

The metric sweeps over a range of IoU thresholds, at each point calculating an average precision value. The threshold values range from 0.4 to 0.75 with a step size of 0.05: (0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75). In other words, at a threshold of 0.5, a predicted object is considered a "hit" if its intersection over union with a ground truth object is greater than 0.5.

At each threshold value t, a precision value is calculated based on the number of true positives (TP), false negatives (FN), and false positives (FP) resulting from comparing the predicted object to all ground truth objects:

$$\frac{\mathrm{TP(t)}}{\mathrm{TP(t)+FP(t)+FN(t)}}$$

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.

Important note: if there are no ground truth objects at all for a given image, ANY number of predictions (false positives) will result in the image receiving a score of zero and being included in the mean average precision.

The average precision of a single image is calculated as the mean of the above precision values at each IoU threshold:

$$\frac{1}{|Thresholds|}\sum_t \frac{\mathrm{TP(t)}}{\mathrm{TP(t)+FP(t)+FN(t)}}$$

Cost/Loss Function - MSE

Object detection neural networks commonly use ℓ1-norm or ℓ2-norm for their cost function. However, there is not a strong correlation between minimizing the loses and improving the IoU (Rezatofighi, 2019).

We plan to use Mean squared error (MSE) or Binary Cross Entropy as the cost/loss function for the models that we will explore.

Model selection and considerations

Currently we have worked on baseline model with following structure and following accuracy, hence as shown above we will try to improve on our model by using "transfer learning", to achieve our stated objective.

Our research points out that two other models YOLO and UNET can also give higher mean average precision at a given IOU threshold as compared to CNN based models, hence we plan to try these models also going forward.

Our research also indicated that use of different CNN networks as backbone can also improve recall value of a model, hence might use that approach also in deciding final model architecture.

The below data is from a research paper and shows how the various CNN models perform on same pneumonia detection data set (Zhenjia Yue)

Model name	Accuracy	Recall
MobileNet	0.92986	0.98984
ResNet-18	0.85515	0.98947
ResNet-50	0.87486	0.98531
VGG19	0.90529	0.78635
CNN	0.91446	0.98813

Model 1 – Baseline Model – CNN – Sequential.

Ref: 01_Group-8_Nov-B_Batch_Computer_Vision_Capstone_Basic_Model_for_Final_Report.ipynb

We have built a baseline model which is a CNN model.

About the model

The CNN model is a sequential deep learning model. The model has been built to using Conv2D and Relu activation function in the top layers. We have used 2x2 pooling and batch normalization. The model uses 0.5 dropout for regularization and preventing overfitting. The model has a fully connected dense layer with 64 neurons followed by a single neuron in the tail of the model. Finally, we have used Sigmoid activation in the last layer to convert the model's output into a probability score, which can be easier to work with and interpret classification. We have used Adam optimizer with a learning rate of 0.0001 and Binary Cross Entropy as the cost/loss function.

The model generates, 422,033 parameters of which 421,809 are trainable and 224 Non trainable.

We fitted the model and set it to run 15 epochs and got loss: 0.60 - acc: 0.90

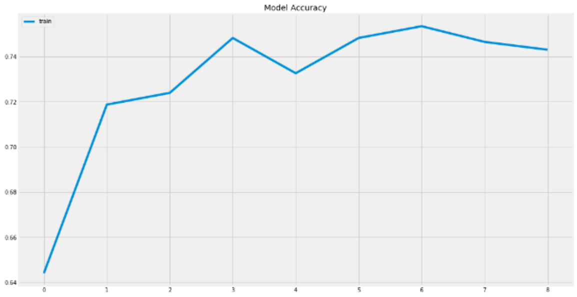
The performance metrics and description of our base line model are as follows: -

Table 1

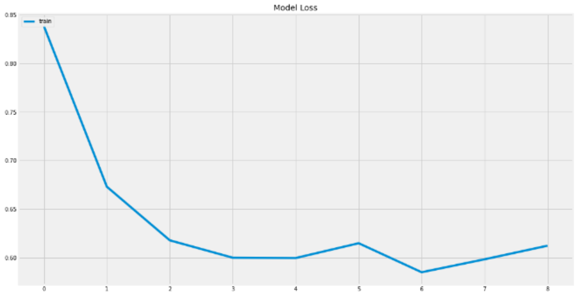
Model Type	CNN -Sequential
Accuracy	0.90
Precision	0.94
Recall	1.00
F1 Score	0.94
AUC Score	0.48

Model Performance

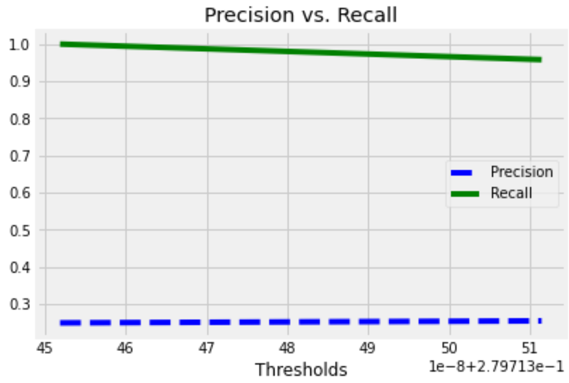
Model Accuracy



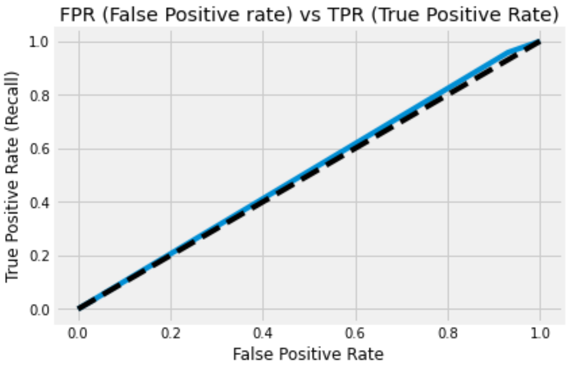
Model Loss



Model Precision vs. Recall



Model FPR vs. TPR



Way Forward for baseline model

Our model has relatively lower AUC Score and Accuracy. Further fine tuning is planned with data balancing and image resizing. In our opinion these measures will help improve the model performance.

Our research indicates the model performance can be further improved by “Transfer Learning” approach.

CNN with Resnet50 backbone

Ref: 02_Group-8_Nov-B_Batch_Computer_Vision_Capstone_Resnet50_Pneumonia_Detection.ipynb

About the model

The second model we built is based on Resnet50 backbone for prediction and using SVC model for classification as a prerequisite.

Our Approach

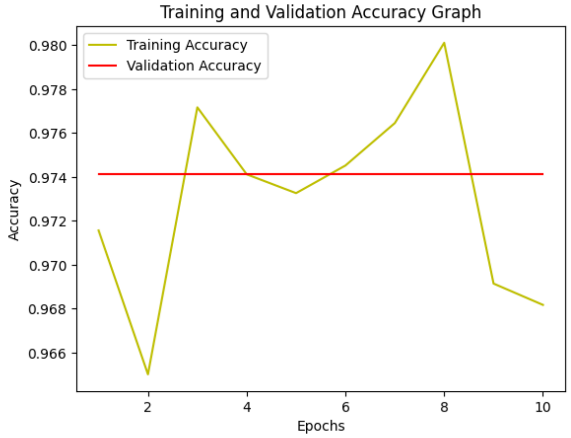
For this model we changed our approach for working with the large image data. We used a Class generatortransfer to create a generator that loads data on the fly. The data generator takes filenames, batch_size and other parameters as inputs and creates a random batch of numpy images and numpy masks.

The model uses Resnet50 as a backbone and transfer learning concept for reducing the training time. The model is compiled using Adam optimizer, Binary cross entropy loss and the performance is measured by accuracy and mean IoU.

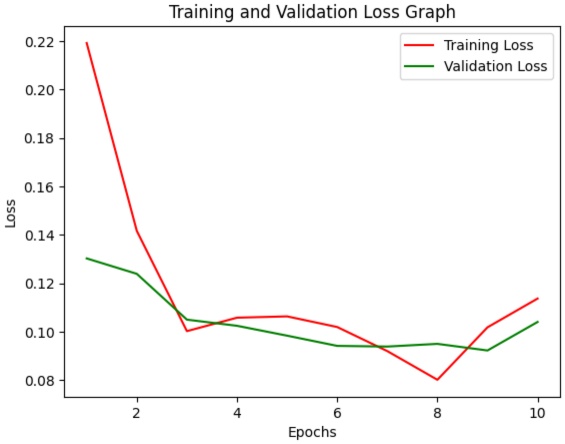
The model performs with a loss of 0.1043 and accuracy of 0.9718 on training data.

On test data the model performs with a loss: 0.0896 - accuracy: 0.9790 - mean_iou: 0.7815 - val_loss: 0.1106 - val_accuracy: 0.9718 - val_mean_iou: 0.7664 when executed for 10 epochs.

Accuracy Graphs



Loss graph



Mask RCNN

Ref: 03_Group-8_Nov-B_Batch_Computer_Vision_Capstone_MaskRCNN_Pneumonia_Detection.ipynb

About the model

We understood that Mask RCNN, aims to solve the instance segmentation problem in computer vision. The model separates different objects in an image (Zhang, 2018). It is a region proposal model and work in two steps; Step 1 – generating proposals of regions in the images where the objects may be located. Step 2 – I predicts it predicts the class of the object, refines the bounding box, and generates a mask in pixel level of the object based on the first stage proposal.

The model is based in the Feature Priority Network (FPN) and Resnet50 backbone.

Illustration of the model is given below (Zhang, 2018): -

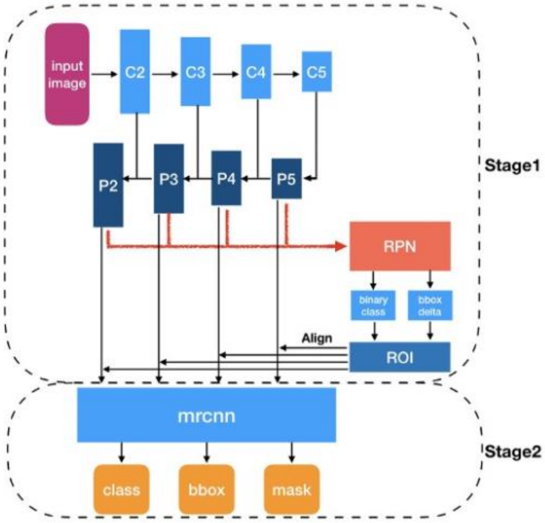


Illustration of Mask RCNN structure

Our approach

We have tried transfer learning from Mask R-CNN built on FPN and ResNet50.

We have used data generator to feed the images to the model for training. The data generator helps in augmentation of the images as well. The data generator uses the flip-lr and adds gaussian noise i.e. blurr and sharpness to the images.

While transfer learning helped a lot to train the model on custom dataset, the team understood that the training could be much quicker with more hardware support. Training our model using Google Colab on Tesla T4 GPU took 7.5 hours.

The model performance was evaluated based on the MS COCO metrics for Average Precision (AP). The benchmark performance for Mask RCNN model is 0.60 mAP with IoU threshold of 0.5. Our model gives a 0.48 mAP with IoU threshold of 0.5.

We could further enhance our model performance by increasing the training epochs and running model on multiple GPUs. We may also get better performance if we balance the data or putting class weights while training.

YOLO v3 based single shot detection model

Ref: - 04_Group-8_Nov-B_Batch_Computer_Vision_Capstone_Yolov3_Pneumonia_Detection.ipynb

About the model

YOLO means You Only Look Once. YOLO v3 is an incremental improvement of the original YOLO model. We choose YOLO v3 since our desk research indicated that this model is much faster than any other SOTA models. We also understood that the Other algorithms (Mask R-CNN, UNet, FCN, ...) take much longer to train and require more GPUs and parameter tuning.

Our Approach
We have tried transfer learning from YOLO v3 available at
https://github.com/AlexeyAB/darknet.git

Setting up the data for running this model is tedious and one needs to get the directories set up correctly. Creating the training data for the use of this module is quiet an expensive step. The configuration file of the model had to be updated to include the path to training and test data in form of .txt files.

We had to create directories for images, labels, configuration, and metadata. This set-up is extremely important for the execution of model. The source dicom images were converted to .jpg images as needed for model training.

For training we have used the pre trained weights for the model (darknet53.conv.74)

YOLOv3 needs .txt file for each image, which contains ground truth object in the image that looks like

<object-class_1> <x_1> <y_1> <width_1> <height_1>

<object-class_2> <x_2> <y_2> <width_2> <height_2>

We also realized that:

- <object-class>: Since RSNA task is binary classification basically, <object-class> is 0.

- <x>, <y>: Those are float values of bounding box center coordinate, divided by image width and height, respectively.
- <w>, <h>: Those are width and height of bounding box, divided by image width and height, respectively.
- It is different from the format of label data provided by Kaggle.

We created the list of image paths to YOLO. This was needed as YOLO looks at individual images. We created two separate list in text files for training images (5410) and validation images (602).

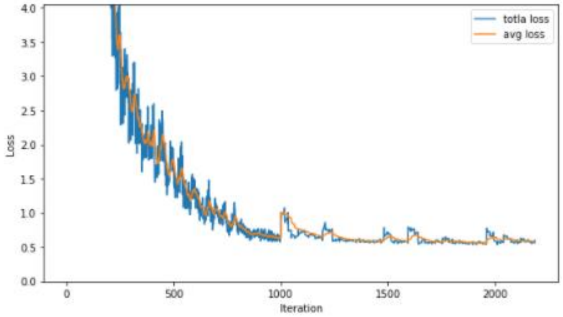
For the model to work the configuration file was modified to point to The RSNA data.

For training the model we used the pre-trained model weights(darknet53.conv.74) . The original YOLO v3 paper indicated that draknet53 to be 1.5X faster than Resnet-101 and performs better as well. Darknet 53 has 53 convolution layers.

Based on the issues explained in darknet GitHub issue, the configuration file was modified to match the need for this project. We adjusted the filter by keeping classed as 2 and num as 9. We kept 21 filters.

The training loss graph of the model is given below*. This is for a set of 2000 iterations. We experienced extremely high loss at the beginning.

* the training loss graph in not included in the .ipynb file as we took it from one of our training files.



We used the pretrained weights on 15300 iterations. The average precision for this model is 85.10% for threshold of 0.25. The precision is 89% and recall of 78% and F1 Score of 0.83. with the IoU threshold = 50 % and using Area-Under-Curve for each unique Recall. Mean average precision (mAP@0.50) = 0.851040, or 85.10 %

Image detection time: .091 seconds per image. In our observation is the faster than the other models we have used in this project.

Strategies for improving model performance

Image enhancements

To further enhance our model performance, we tried balancing the data and image resizing. However, since this led to varied results in the model and since we were working with medical data the team decided to not enhance the data for model training purposes. Only image resizing done for displaying the images in our outputs.

Image Augmentation

Image augmentation techniques such as manipulation of brightness, contrast and image blurring was done for Mask RCNN.

For YOLO v3 based model the images were converted to .jpg format to make the images compatible with the model.

Summary of model performances

We have summarized the model performances in the table below. All the models were run on Tesla T4 GPU with 16 GB RAM and CPU with 12 GB RAM. It is evident to us that while the models were trained using single GPU, the training can be substantially faster if we use multiple GPUs.

Table 2

Model name	Parameters	No of Epochs/ Iterations	Accuracy	F1	Recall	Precision	mAP (IoU)
Baseline Model	422,033	15	0.9	0.94	1.00	0.94	NA
Resnet-50	26,358,993	10	0.97	-	-	-	0.77 (0.5)
Mask RCNN	44,662,942	16	-	-	-	-	0.48 (0.5)
Yolo v3	Total BFLOPS 139.496	15300	-	0.83	0.78	0.89	0.85 (0.5)

BFLOPS – Billions floating point operations per second.

Conclusions

We see from the above table that each model has merits and de-merits. Models like Mask RCNN offer better accuracy at the cost of more time they take to provide the predictions. While the model like YOLO v3 offers greater speed of predictions but relatively less accuracy. The comparison of validation loss of these models also indicates a favorability to YOLO v3. The detection of Pneumonia and putting bounding box around the patch in CXR can be very well done by YOLO v3 model. However, if the use case demands more precision then we would suggest use of Mask RCNN.

As a team we concluded that the use of a particular model needs to be determined by the use case for which it is applied.

Our Reflections

1) Working in a team with team members of diverse professional background was rather refreshing experience.
2) Working on the project and balancing professional and personal commitments was managed well by all the team members.
3) Process the DICOM files and understanding the DICM format was new experience for all.
4) The team learnt exploratory data analysis for DICOM images.
5) Python skills were tested and enhanced due to this project.
6) We explored the Keras, Tensorflow and the use of these by various researchers and competition participants.
7) It was quite a revelation that few models could best run within a specific environment set-up.
8) Few of us from the non-technical background also learnt the importance of setting-up directories and impact of not setting the directories correctly.
9) We also experienced the technicalities involved in using learned weights for models like YOLO v3 and Mask RCNN
10) Last but not the least we learnt about mAP and setting the IoU threshold.

Our way forward

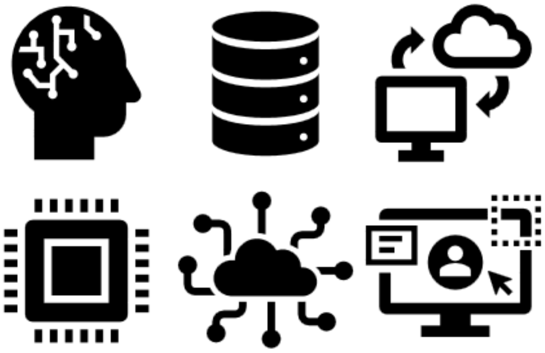
We perhaps would plan to design a web based system that analyzes the DICOM images and our model is built in this system. The model then classifies the CXRs as

1) Pneumonia
2) Other abnormality
3) Normal

For the CXRs classified as CXR with Pneumonia, our model will put a bounding box indicating the location of abnormality.

The detection system will comprise of a UI to be used by health care providers and nurses. These end users will upload a CXR image and the model will provide the diagnosis along with the bounding box placed on the CXR in case the diagnosis is “Pneumonia”. The system will provide option to the end user to accept the diagnosis as is or pass it on to the medical professional for further confirmation. The outcome of this “adjudication” process is recorded in the system.

The input provided by end user is processed by the model and output relayed to the user. The images fed to the model are stored in the data layer of this application.

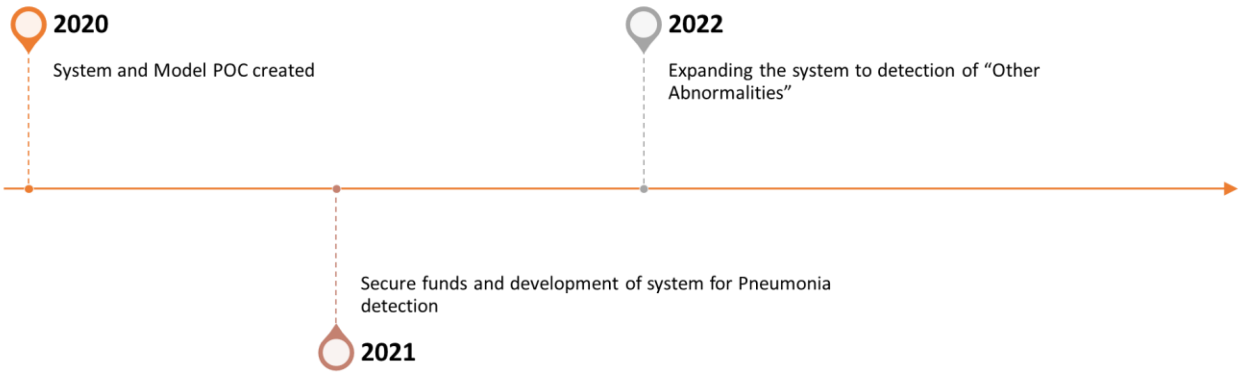


Solving the business need

The key pain area for health care professionals worldwide is the time at hand to attend the patients and to spend on diagnosis. The system that we will develop will reduce the diagnosis time to bare minimum. This will significantly improve the time a healthcare provider spends with the patient.

Our way forward for System/Product Development

Our short term and long term goals are given below. If supported by adequate resources, we would want to develop our model further in 2021 with an aim to productionize it by end of 2021.



2020

System and Model POC created

2021

Secure funds and development of system for Pneumonia detection

2022

Expanding the system to detection of “Other Abnormalities”

Points to consider

Business perspective

We are working on following points.

1) The viability of this product in Indian healthcare system
2) The model performance – How much mAP/IoU is enough? In our discussions as a product design team we have understood: -
    a. higher mAP/IoU values may be needed where the diagnosis must be followed by surgical operation e.g. Tumors, malignancies etc.
    b. For diagnosis that has a standard of care which is Non-Invasive, a lower mAP/IoU is

fine but the model should be tuned for accuracy of the classification.

3) The product/system is by no means a replacement for a trained radiologist, however, has a potential to minimize radiologist’s work and revolutionize radiology practice

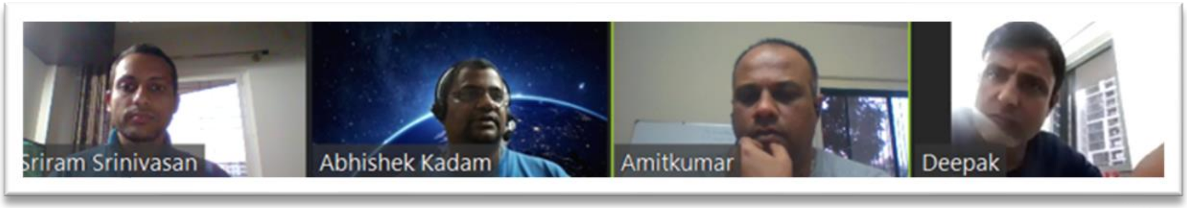
Vote of Thanks

The project team want to sincerely thank our Mentor – Mr. Shirish Patil for the discussion on this project and the guidance and push to complete this project and during all the mentorship period.

The project team want to sincerely thank Program manager – Ms. Shabana Khan for the coordination and streamlining the mentoring sessions.

We also want to thank Great Learning team for the overall support extended to us as we went ahead from one course module to other.

Project team



References

Abbas, A. A. (2020). *Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network. Appl Intell (2020).* Springer.

Arjun_Kashyap. (2019, Dec 02). *Precision and recall — a simplified view*. Retrieved from https://towardsdatascience.com/: https://towardsdatascience.com/precision-and-recall-a-simplified-view-bc25978d81e6

https://stoppneumonia.org/wp-content/uploads. (2019, 11 12). Retrieved from https://stoppneumonia.org: https://stoppneumonia.org/wp-content/uploads/2019/11/India-12.11.2019-Web.pdf

Rezatofighi, H. a. (2019). *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

stoppneumonia. (2019, 11 12). *https://stoppneumonia.org/wp-content/uploads/2019/11/.* Retrieved from https://stoppneumonia.org: https://stoppneumonia.org/wp-content/uploads/2019/11/India-12.11.2019-Web.pdf

Yudong Liu, Y. W. (2019). *CBNet: A Novel Composite Backbone Network Architecture for Object Detection.* arxiv.org.

Zhang, X. (2018, Apr 23). *Simple Understanding of Mask RCNN.* Retrieved from medium.com: https://alittlepain833.medium.com/simple-understanding-of-mask-rcnn-134b5b330e95#:~:text=Mask%20RCNN%20is%20a%20deep,two%20stages%20of%20Mask%20RCNN.

Zhenjia Yue, L. M. (n.d.).