FDA Submission

Your Name: Ahmed Okasha

Name of your Device: Pneumonia Diagnosis Algorithm

Algorithm Description

1. General Information

Intended Use Statement: Assisting radiologists in detecting Pneumonia.

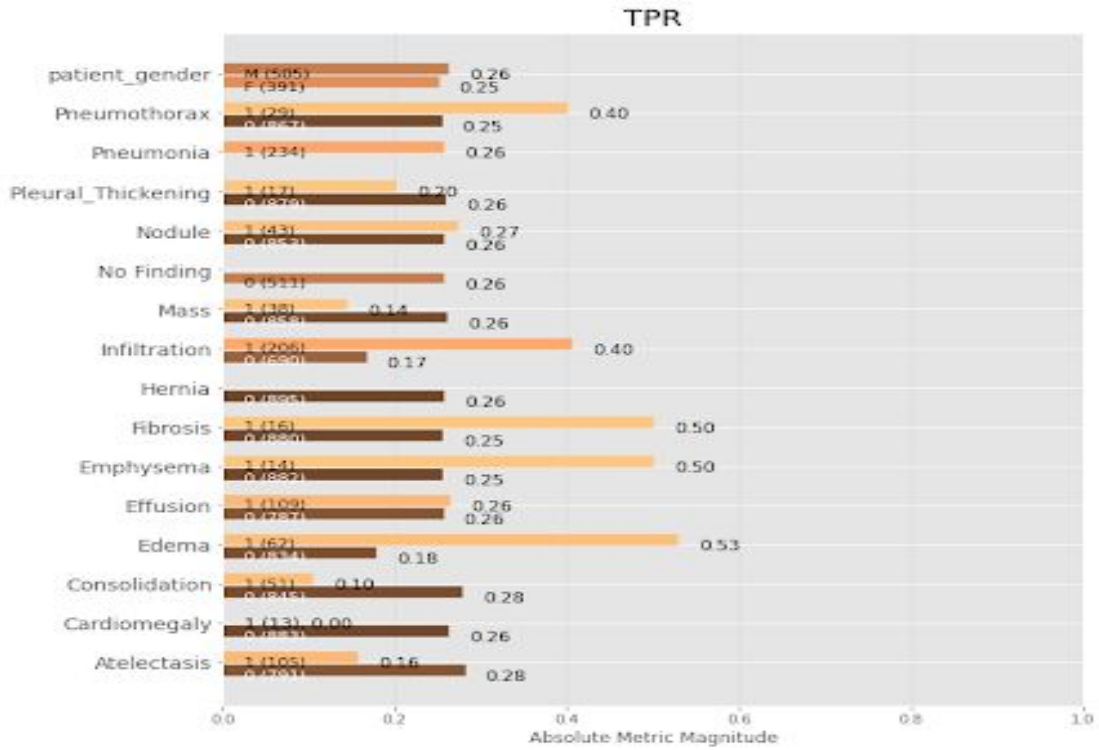
Indications for Use:

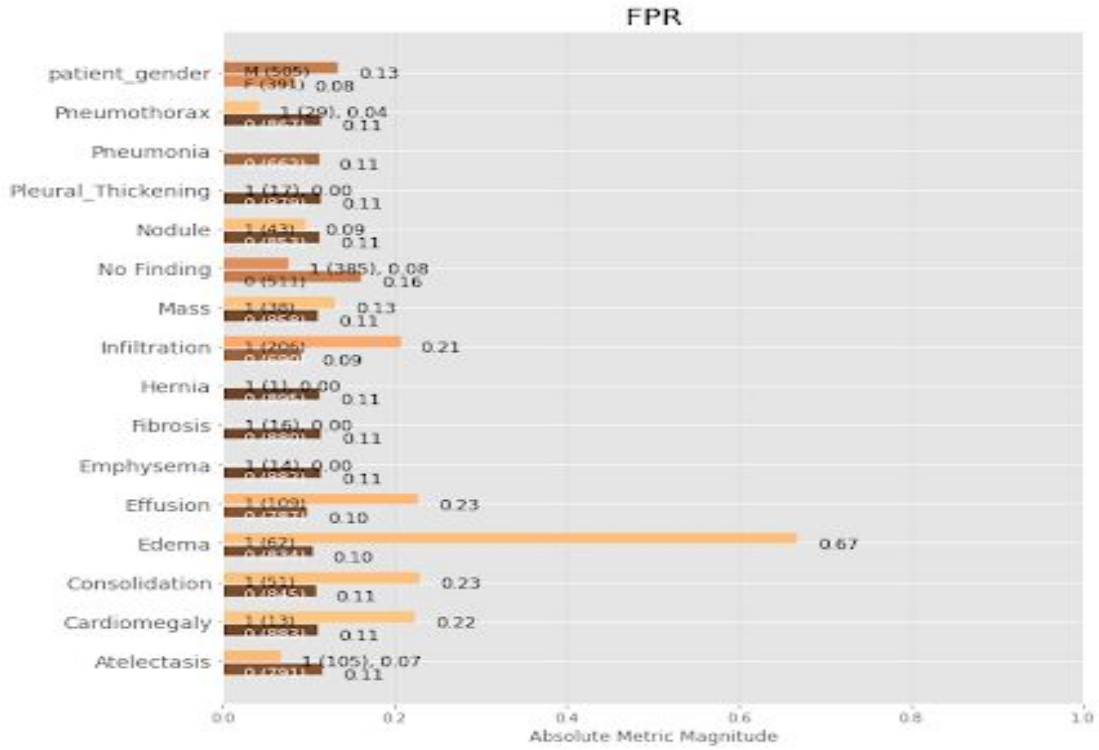
Screening x-ray images on AP, PA positions to detect the presence of Pneumonia. This algorithm is intended for use on both males, females from the ages of 20-70 with a follow-up history between 0-8 and with a number of 0-3 lung issues.

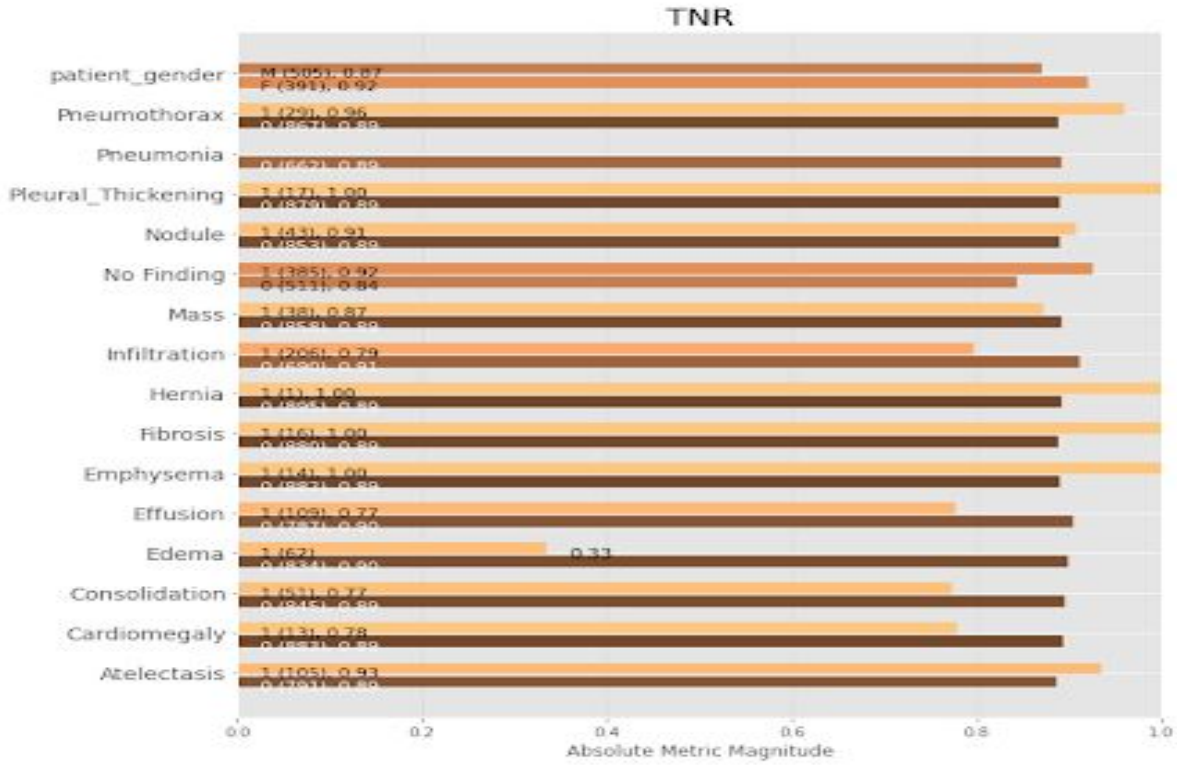
Device Limitations:

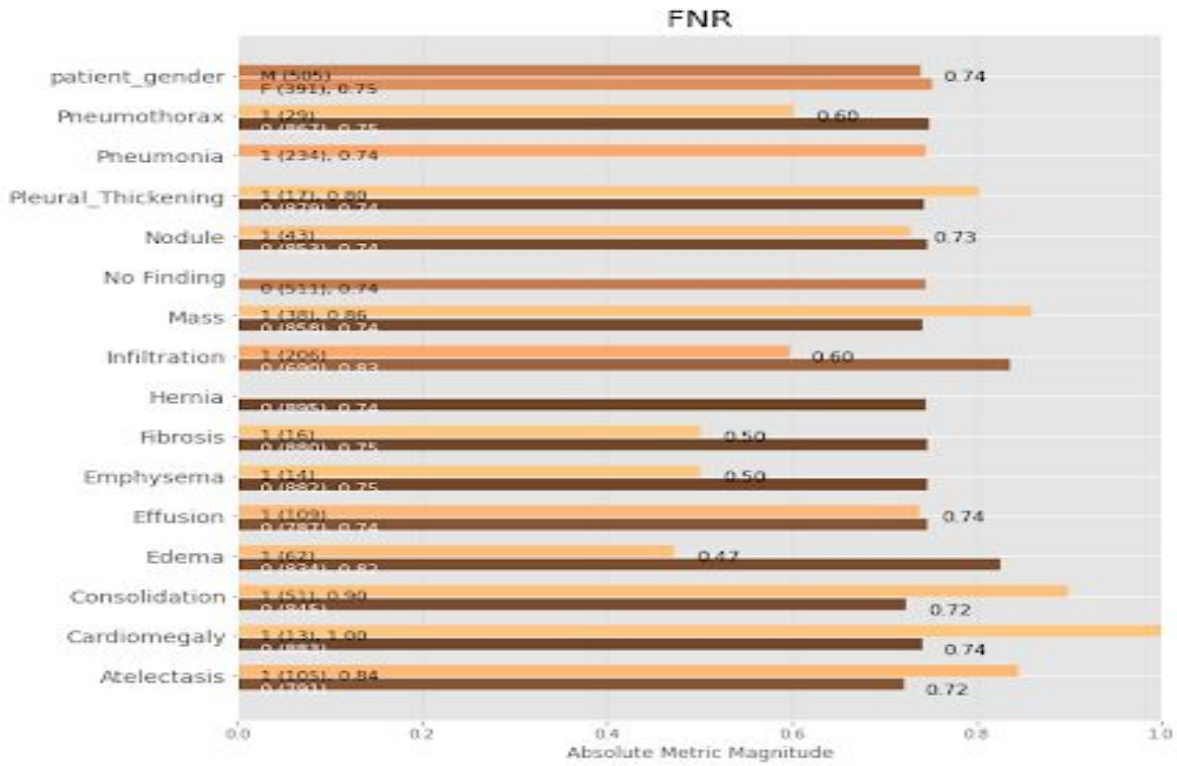
Algorithm Limitation: The statistical bias analysis for the algorithm shows:

- The true positive rates noticeably increase with the existence of Infiltration, Edema.
- False Positive rate with the existence of Pneumonia increases with the existence of Infiltration, Edema, and Efusion
- The false negative rate increases with the existence of Hernial, Feprosis, Emphysema, Edema


TPR
patient_gender
Pneumothorax
Pneumonia
Pleural_Thickening
Nodule
No Finding
Mass
Infiltration
Hernia
Fibrosis
Emphysema
Effusion
Edema
Consolidation
Cardiomegaly
Atelectasis
0.0
0.2
0.4
0.6
0.8
1.0
Absolute Metric Magnitude


FPR
patient_gender
Pneumothorax
Pneumonia
Pleural_Thickening
Nodule
No Finding
Mass
Infiltration
Hernia
Fibrosis
Emphysema
Effusion
Edema
Consolidation
Cardiomegaly
Atelectasis
0.0
0.2
0.4
0.6
0.8
1.0
Absolute Metric Magnitude


TNR
patient_gender
Pneumothorax
Pneumonia
Pleural_Thickening
Nodule
No Finding
Mass
Infiltration
Hernia
Fibrosis
Emphysema
Effusion
Edema
Consolidation
Cardiomegaly
Atelectasis
0.33
0.0
0.2
0.4
0.6
0.8
1.0
Absolute Metric Magnitude


FNR
patient_gender
Pneumothorax
Pneumonia
Pleural_Thickening
Nodule
No Finding
Mass
Infiltration
Hernia
Fibrosis
Emphysema
Effusion
Edema
Consolidation
Cardiomegaly
Atelectasis
0.74
0.60
0.73
0.60
0.50
0.50
0.74
0.47
0.72
0.74
0.72
0.0
0.2
0.4
0.6
0.8
1.0
Absolute Metric Magnitude

Computational Limitation:

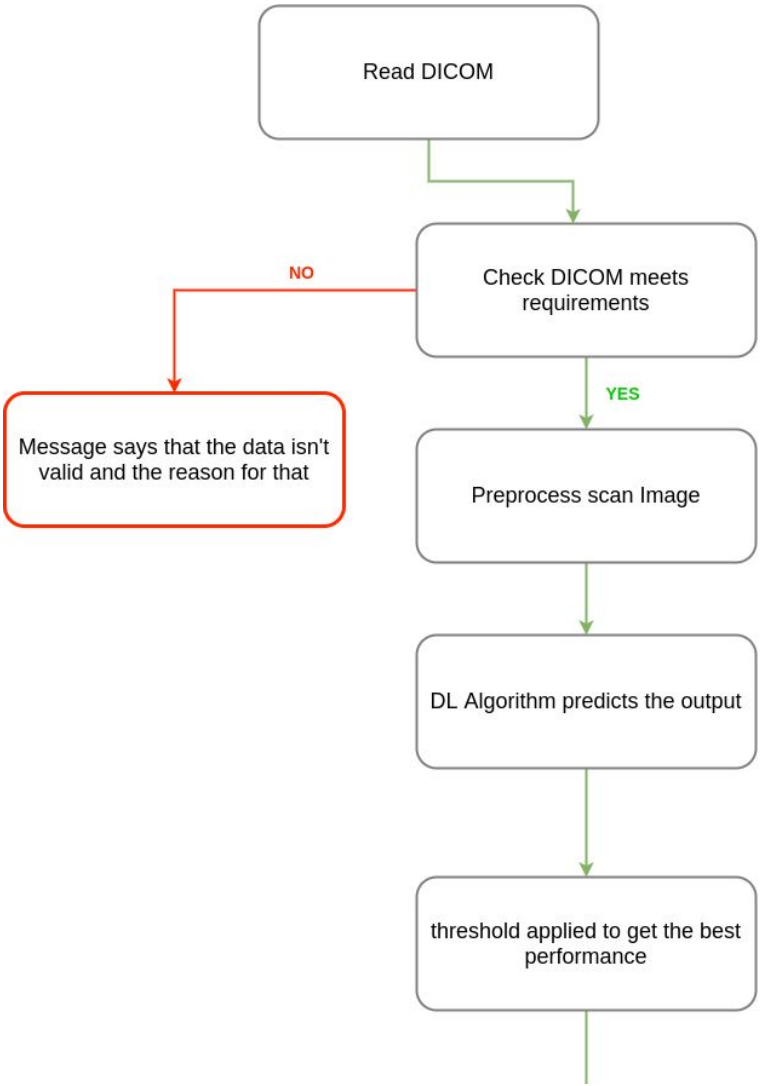
The algorithm can run on a device without and gives inference time of ~0.5s, with GPU CUDA devices this time can be reduced to ~40ms.

Clinical Impact of Performance:

It can save a lot of time in checking many radiographs x-rays for pneumonia. Gives a higher overall f1 score than radiologists in detecting pneumonia. This algorithm can be used to assist radiologists for prioritizing scans with high pneumonia probability over other findings which is useful in many situations like COVID-19.

there is many radiograph findings that can be detected nowadays, this high number of findings makes radiologists task more difficult than ever and makes the overall sensitivity, specificity low in many findings, so an algorithm which specializes in detecting certain finding like Pneumonia can be very helpful.

2. Algorithm Design and Function



DICOM Checking Steps:

1- DICOM files should be for patients within the correct age range

2- DICOM files should be for the correct positions algorithm trained on.

3- x-rays images in the DICOM file is resized, reshaped to the correct model input which is (1,224,224,3)

Preprocessing Steps:

A- For training Process:

1-image pixel values are scaled to be in the range of 0-1

2- image is resized and reshaped to be with shape of (1,224,224,3)

3-Augmentation like: random rotation width shift, height shift, brightness range, sheer range zoom range horizontal flip is added

B- For testing and validation process :

1-image pixels is scaled to be in the range of 0-1

2- image is resized and reshaped to be with shape of (1,224,224,3)

CNN Architecture:

The architecture used in the Algorithm is a VGG16 model with ImageNet pre-trained weights, all layers from the beginning were chosen until “block5_pool” layer removing the following layers to the end. all the chosen layers from the start were frozen so training will not update their weights until “block5_conv2”.

After modifying VGG16 model, these layers were added:

1- Flatten() to enable the output of the maxpool layer to be fed to the Dense layers

2- Dense(128,activation="relu",kernel_regularizer=K.regularizers.l2(0.001)) fully connected layer with 128 unit and l2 regularizer is used to avoid overfitting

my_model.add(BatchNormalization()) dynamic normalization technique for the weights of the layer towards the direction that increases the accuracy.

3- Dense(64,activation="relu",kernel_regularizer=K.regularizers.l2(0.001))

4- BatchNormalization()

4- Dense(16,activation="relu"))#kernel_regularizer=K.regularizers.l2(0.4)))

5-Dense(1,activation='sigmoid')) gives an output from 0-1 that can be converted to the binary output

3. Algorithm Training

Parameters:

1- Types of augmentation used during training

- rotation_range=5 (add random rotation between 0-5 angles)
- width_shift_range=0.01(add random shift in width between 0-0.01)
- height_shift_range=0.01(add random shift in height between 0-0.01)
- brightness_range=[0.05,0.1](add random brightness in range between 0.05-0.1)
- shear_range=0.01(add random shear in range between 0-0.01)
- zoom_range=0.01(add random zoom in the range between 0-0.01)
- horizontal_flip=False(should be true in normal cases, but the accuracy was better in the validation set without it)
- vertical_flip=False,
- validation_split=0.0 (not to split the training dataset because we have already validation dataset)

2- Batch size

Chosen batch size=128 for both the training an validation step

3- Optimizer learning rate

Learning rate was set to 5e-3 and with ReduceLROnPlateau() callback to reduce the learning rate when the loss reaches plateau over epochs

Adam optimizer was used to

4- Layers of pre-existing architecture that were frozen

All layers after “block5_pool” were removed from the model.

5- Layers of pre-existing architecture that were fine-tuned

All layers starting from “block5_conv2” were set to be trainable which are these layers “block5_conv2”, “block5_conv3” and “block5_pool” but pooling layers have no trainable parameters anyway.

6- Layers added to the pre-existing architecture

1- Flatten() to enable the output of the maxpool layer to be fed to the Dense layers

2- Dense(128,activation="relu",kernel_regularizer=K.regularizers.l2(0.001)) fully connected layer with 128 unit and l2 regularizer is used to avoid overfitting my_model.add(BatchNormalization()) dynamic normalization technique for the weights of the layer towards the direction that increases the accuracy.

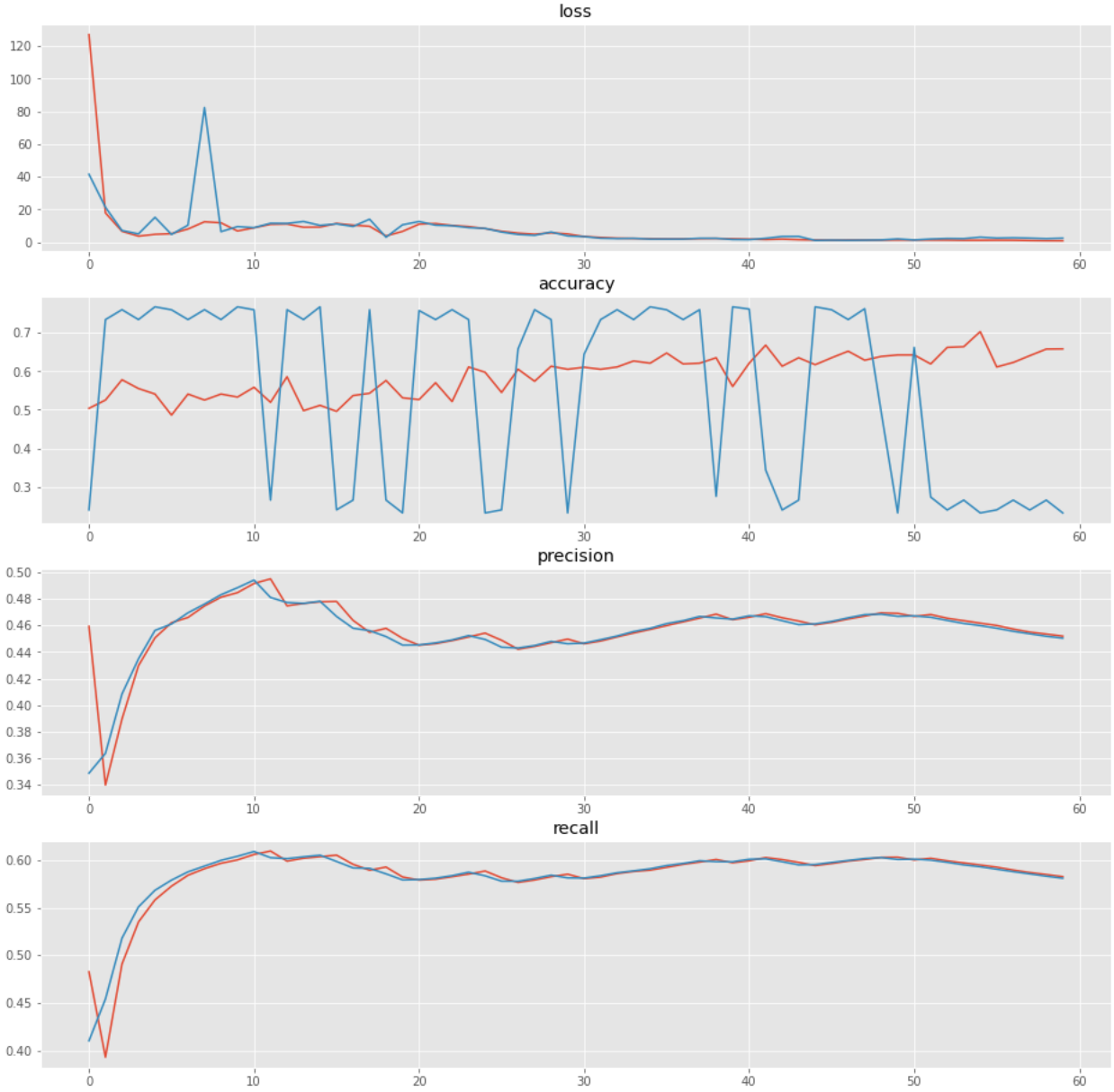
3- Dense(64,activation="relu",kernel_regularizer=K.regularizers.l2(0.001))

4- BatchNormalization()

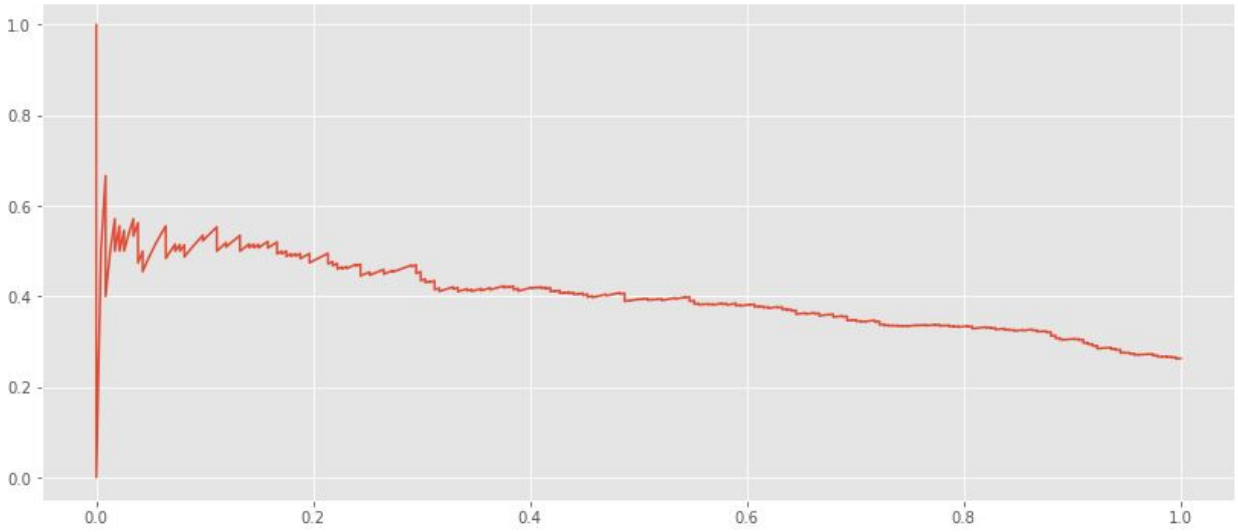
4- Dense(16,activation="relu"))#kernel_regularizer=K.regularizers.l2(0.4)))

5- Dense(1,activation='sigmoid')) gives an output from 0-1 that can be converted to the binary output

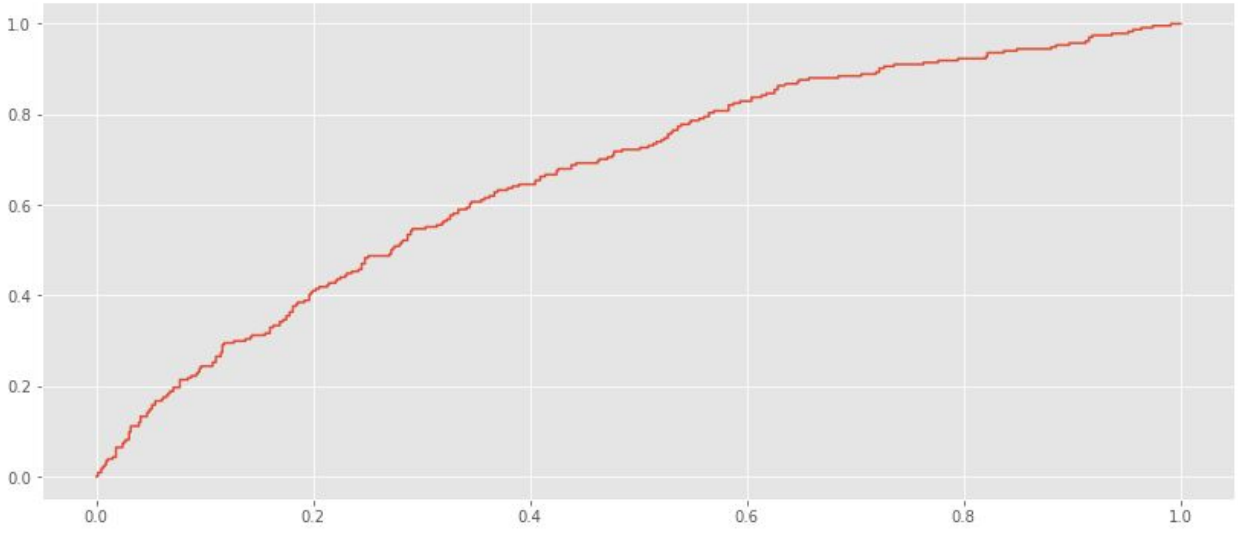
6- Algorithm training performance visualization



7- Insert P-R curve



ROC



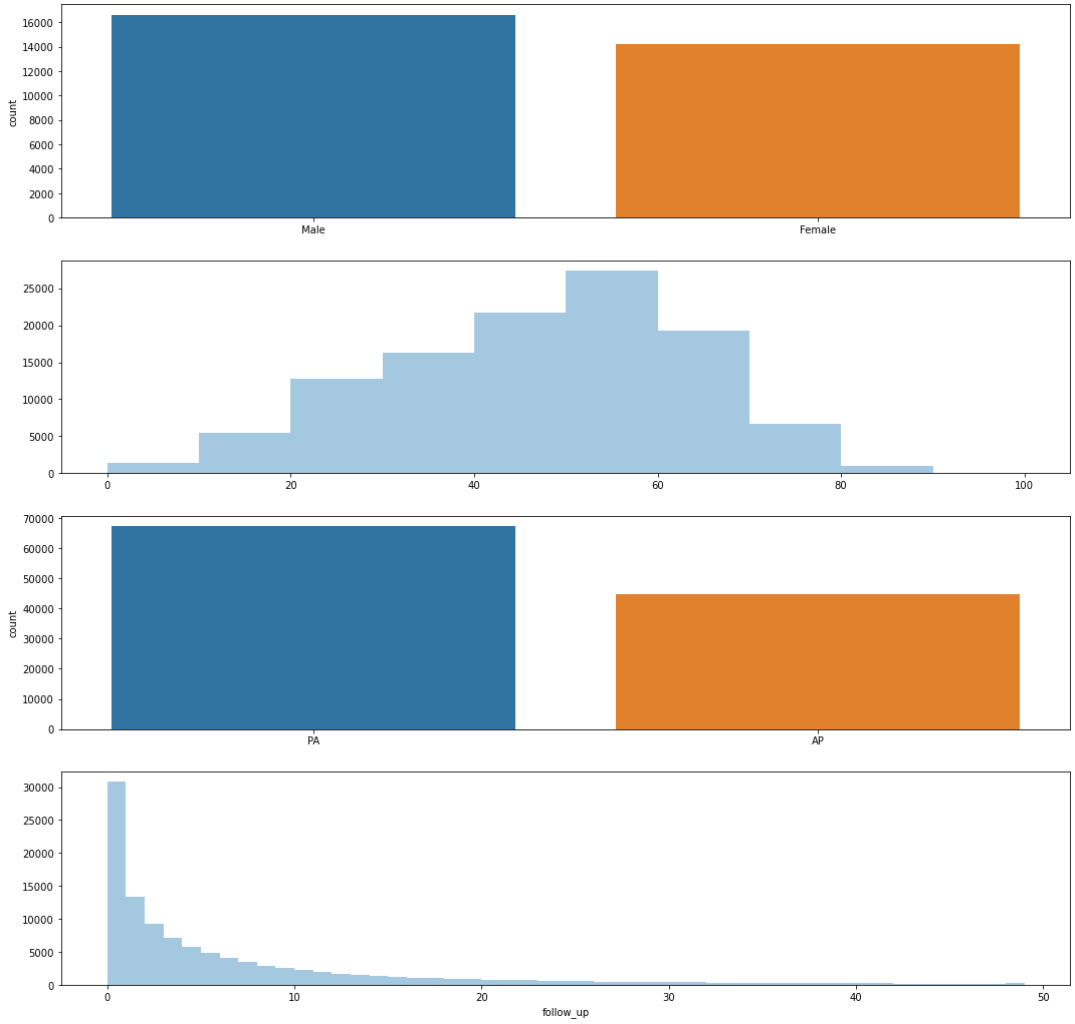
8-Final Threshold and Explanation:

The selected threshold was the threshold that gives the f1-score as high as possible which is 0.935 in our case which gives these statistics over 896 images from the validation dataset with a positive:negative ratio 1:4.
Resulted TPR, FPR, FNR, TNR are 0.25641026, 0.74358974, 0.1102719, 0.8897281

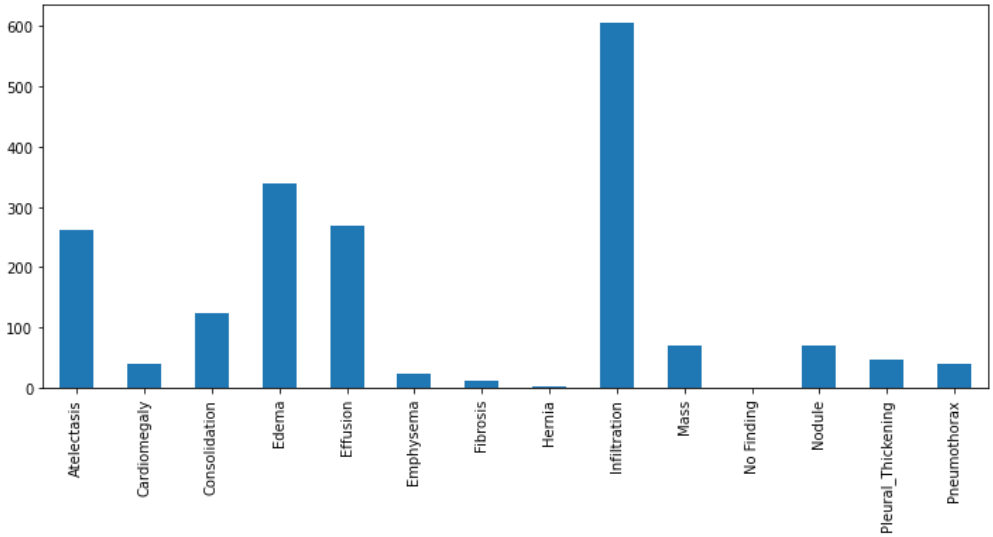
	precision	recall	f1-score	support
0	0.77	0.89	0.83	662
1	0.45	0.26	0.33	234
accuracy			0.72	896
macro avg	0.61	0.57	0.58	896
weighted avg	0.69	0.72	0.70	896

4. Datasets

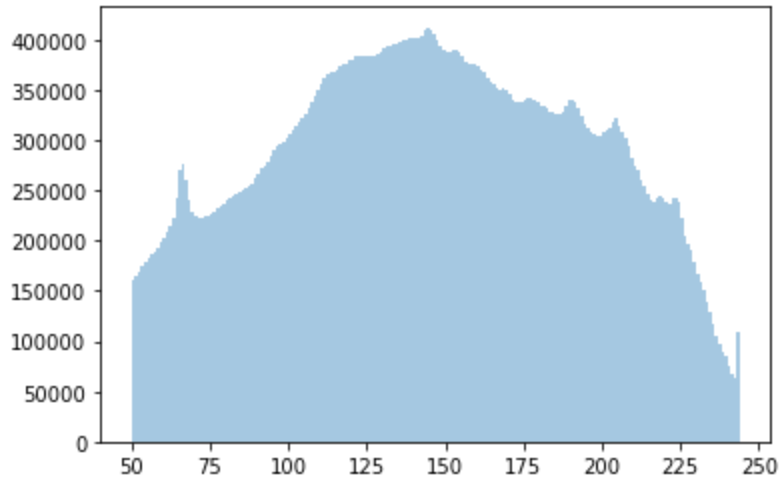
The distribution of labels over the original dataset:



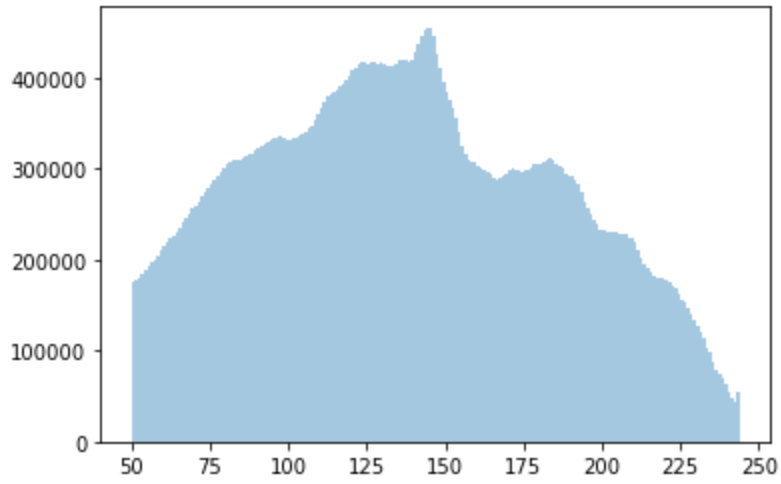
Cooccurrence of the distribution of the findings in the original dataset



Positive Pneumonia scans intensity



Negative Pneumonia scans intensity



0- Description of The whole Dataset:

The used dataset is the NIH Chest X-ray Dataset is composed of 112,120 X-ray images with disease labels from 30,805 unique patients. To create these labels, the authors used Natural Language Processing to text-mine disease classifications from the associated radiological reports. The labels are expected to be >90% accurate and suitable for weakly-supervised learning.

Data Limitations:

1. The image labels are NLP extracted so there could be some erroneous labels but the NLP labeling accuracy is estimated to be >90%.
2. Very limited numbers of disease region bounding boxes (See BBoxlist2017.csv)

Provided Findings:

There are 15 classes (14 diseases, and one for "No findings"). Images can be classified as "No findings" or one or more disease classes:

- Atelectasis
- Consolidation
- Infiltration
- Pneumothorax
- Edema
- Emphysema
- Fibrosis
- Effusion
- Pneumonia
- Pleural_thickening
- Cardiomegaly
- Nodule Mass
- Hernia

Our focus on this project was to detect the Pneumonia existence only and therefore preprocessing for this dataset was required before training.

1- Description of Training Dataset:

The training data set is selected with these points set as the main goal:

1- the “pneumonia class should be balanced across all the dataset”
2- the dataset remaining labels distribution should be the same as the provided dataset to avoid any biases and to make the training set representative
3- there should be no data leakage between both training and validation datasets that can happen because some patients have several scans across the dataset

To achieve these goals these steps were made :

1- the dataset were grouped by “patient_id” label into a new data frame, and all the other columns were aggregated

2- the “pneumonia_class” was changed so that any patient with a positive record will be considered positive at this step only.

3- the new dataset was split between training and validation with a percentage of (0.90,0.10), shuffling was added to make the two datasets representative as possible to the main dataset.

4-the training dataset was de-grouped again so that each column represent one record and the pneumonia class was changed back to represent one scan only.

5- 1,2,4 steps were done mainly to make each patient records in one dataset only.

6- the training dataset was resized to make the “pneumonia” class balanced, shuffling was again to avoid any bias.

Description of Validation Dataset:

4-the validation dataset obtained after splitting the grouped dataset was de-grouped again so that each column represents one record and the pneumonia class was changed back to represent one scan only.

5- the distribution of the “pneumonia_class” in the validation dataset was left unbalanced to be representative of the real-world data.

5. Ground Truth

The ground truth in the dataset is obtained by an NLP algorithm that extracts finding from diagnosis reports.

6. FDA Validation Plan

Patient Population Description for FDA Validation Dataset:

The validation dataset should be collected from x-ray scans from patients with age in between 20-70 and a follow-up history between 0-8, with a number of 0-3 lung issues.

Ground Truth Acquisition Methodology:

Because chest radiograph is universally considered to be the “gold standard” for pneumonia diagnosis, a single radiologist diagnosis will be sufficient to obtain the ground truth.

Algorithm Performance Standard:

From https://stanfordmlgroup.github.io/projects/chexnet/
Radiologists average precision and recall is(.330,0.442) and f1 score =0.387, so this can be considered as the standard for detection pneumonia from radiographs only.