Deep Learning, Spring 2020
Assignment 5 - Part 1

Detecting Coronavirus Infections through Chest X-Ray images

Git Repository Link:
https://github.com/wassam21/BSCS17028_COVID19_DLSpring2020

1.0. Objective

In this assignment we are required to write code for detecting infections such as COVID-19 among X-Ray images.

- Use CNN, pre-trained on ImageNet, to extract basic features from X-Ray images.
- Train the classification layers in order to detect instances of Infected (COVID-19 + Pneumonia) and Normal X-Ray images.
- Fine-tune the entire network to try to improve performance.

1.1. Task 01: Load pretrained CNN model and fine-tune FC layers

In this task we will fine-tune two networks (VGG-16 and ResNet-18) pretrained on ImageNet weights. We freeze all the features layers expect FC layers and then trained the FC layers on our dataset. Following are the result of Task 01.

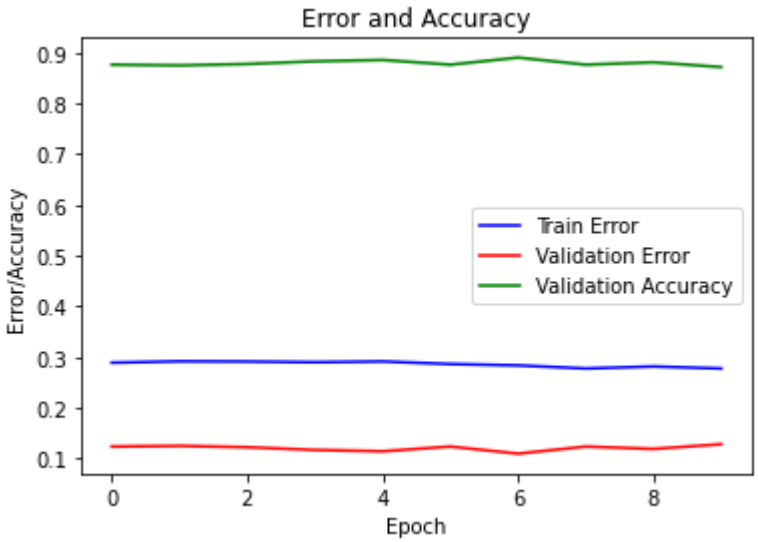
1.1.0. VGG-16

1.1.0.1. Experimental Setup:

Epochs = 10
Learning Rate = 0.001
Feature Layers = Freeze
Fine-tune layers =

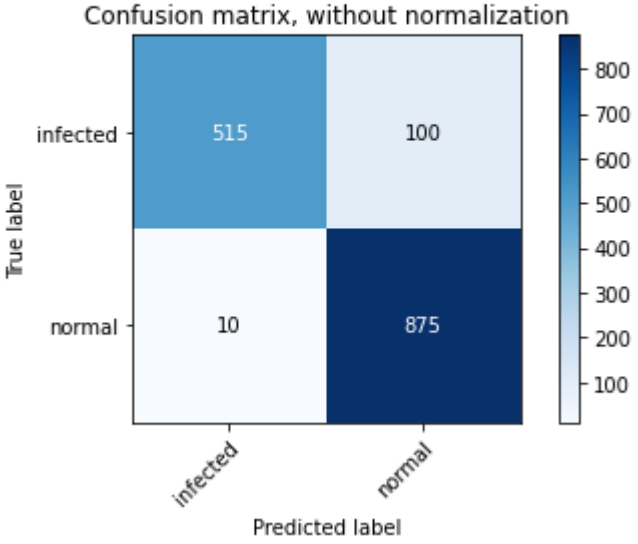
```
(classifier): Sequential(
  (0): Linear(in_features=25088, out_features=380, bias=True)
  (1): ReLU(inplace=True)
  (2): Dropout(p=0.5, inplace=False)
  (3): Linear(in_features=380, out_features=2, bias=True)
)
```

1.1.0.2. Loss and Accuracy Curve:



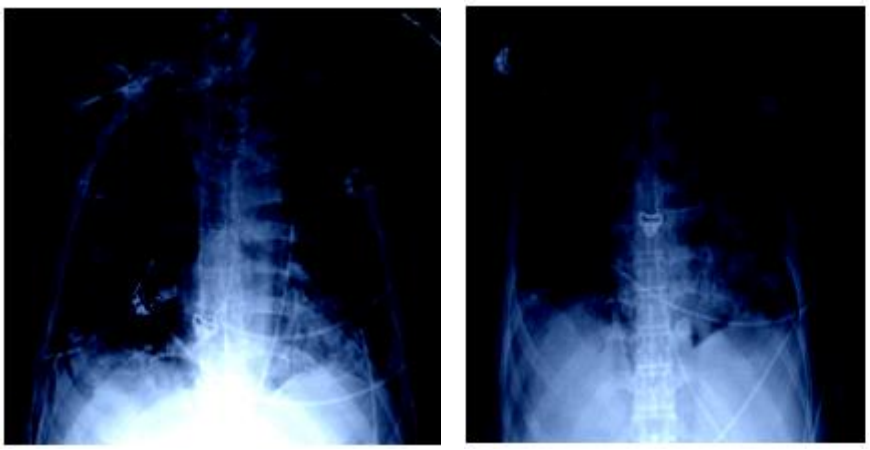
1.1.0.3. Confusion Matrix:

```
Confusion matrix, without normalization
[[515 100]
 [ 10 875]]
```

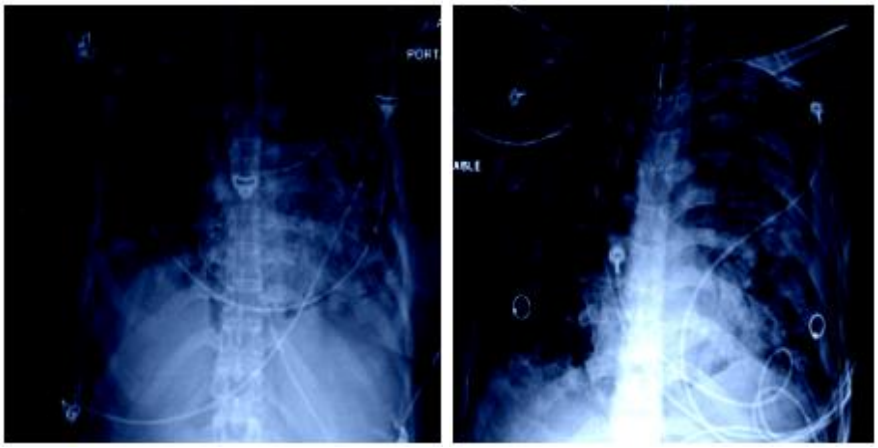


1.1.0.4. Best and Worst Classified Images

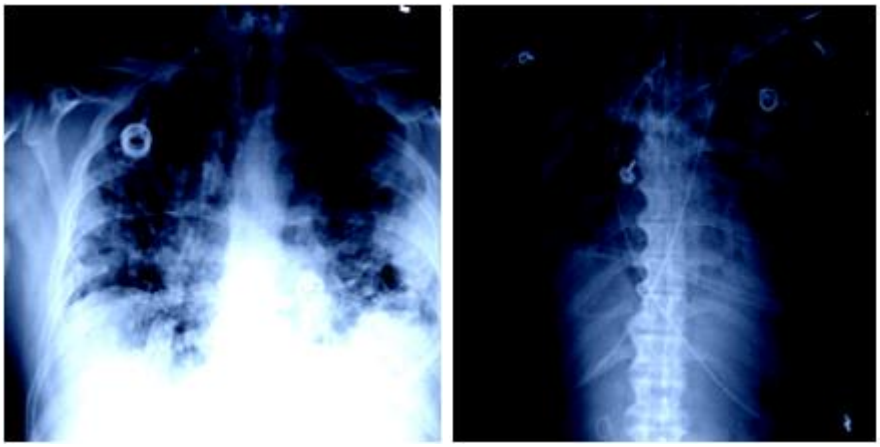
1.1.0.4.0. Best Classified Infected Images



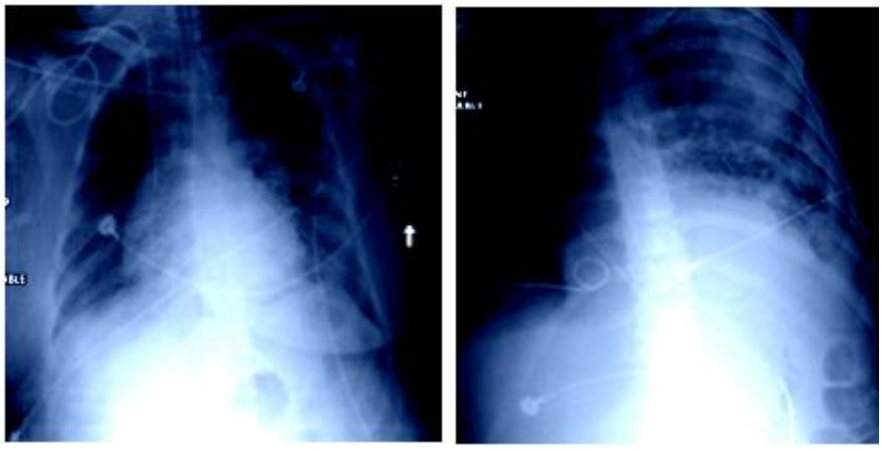
1.1.0.4.1. Best Classified Normal Images



1.1.0.4.2. Worst Classified Infected Images



1.1.0.4.3. Worst Classified Normal Images



1.1.0.5. Final Accuracy and F1 Score:

The final accuracy for test dataset is ***92.6 %*** and F1 score is ***0.94.***

1.1.0.6. Analysis:

The results are very interesting, pretrained weights works fine with this dataset and give very good accuracy of 92.6%. Also, FC layer play its role in the accuracy there are 2 FC layers with 380 and 2 neurons respectively. In confusion matrix you can see False Negative is less than False Positive.

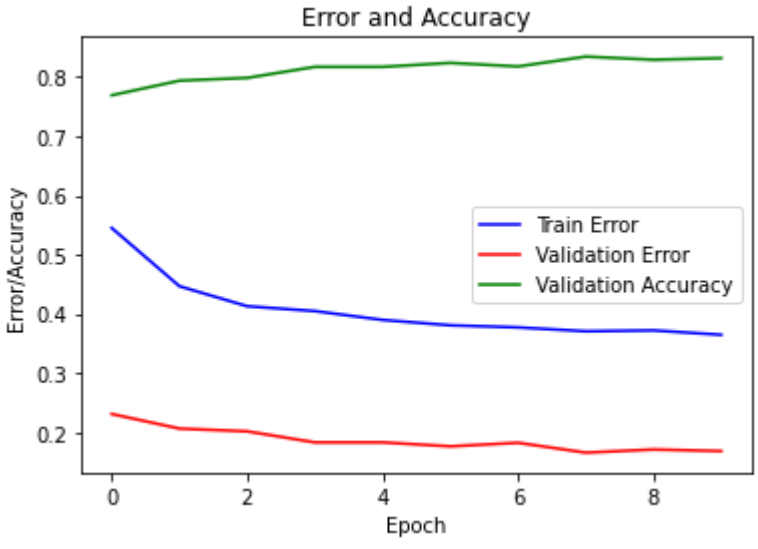
1.1.1. ResNet-18

1.1.1.0. Environment Setup:

Epochs = 10
Learning Rate = 0.001
Feature Layers = Freeze
Fine-tune layers =

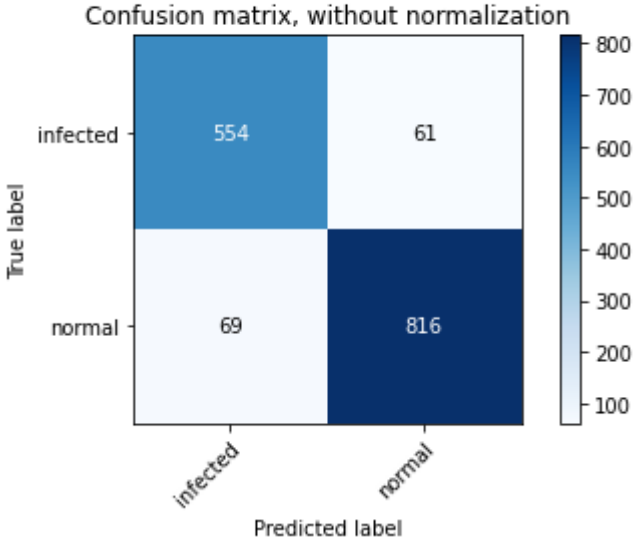
```
Sequential(
  (0): Linear(in_features=512, out_features=380, bias=True)
  (1): ReLU(inplace=True)
  (2): Dropout(p=0.5, inplace=False)
  (3): Linear(in_features=380, out_features=2, bias=True)
)
```

1.1.1.1. Loss and Accuracy Curve:



1.1.1.2. Confusion Matrix:

```
Confusion matrix, without normalization
[[554  61]
 [ 69 816]]
```

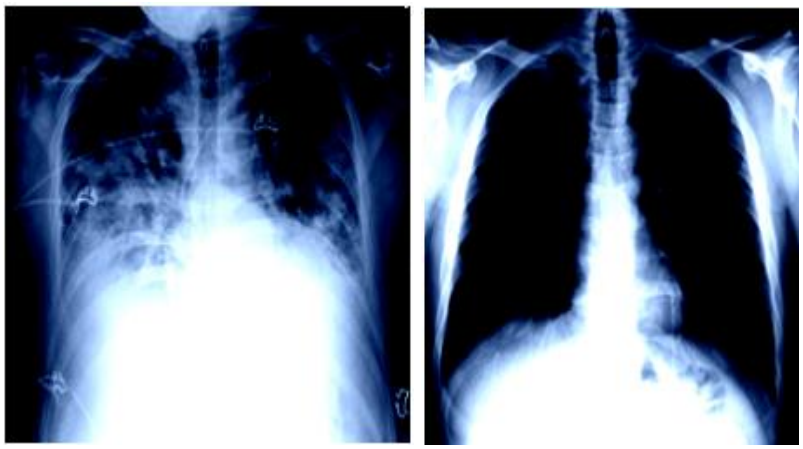


:

1.1.1.3. Best and Worst Classified Images:

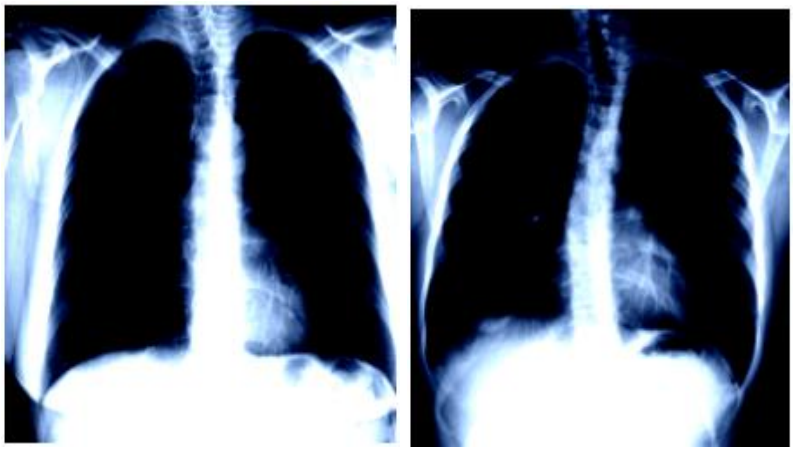
1.1.1.3.1. Best Classified Infected Images:

1.1.1.3.2. Best Classified Normal Images:



1.1.1.3.3. Worst Classified Infected Images:

1.1.1.3.4. Worst Classified Normal Images:



1.1.1.4. Final Accuracy and F1 Score:

The final accuracy for test dataset is ***91.3 %*** and F1 score is ***0.926.***

1.1.1.5. Analysis:

ResNet18 is faster than VGG16 but VGG16 accuracy is slightly high than ResNet18. Both Model run with same experimental setup. In confusion matrix you can see False Negative and False Positive are almost same but in VGG16 False positive are very less compare to ResNet18 but False Positive are greater than Resnet18

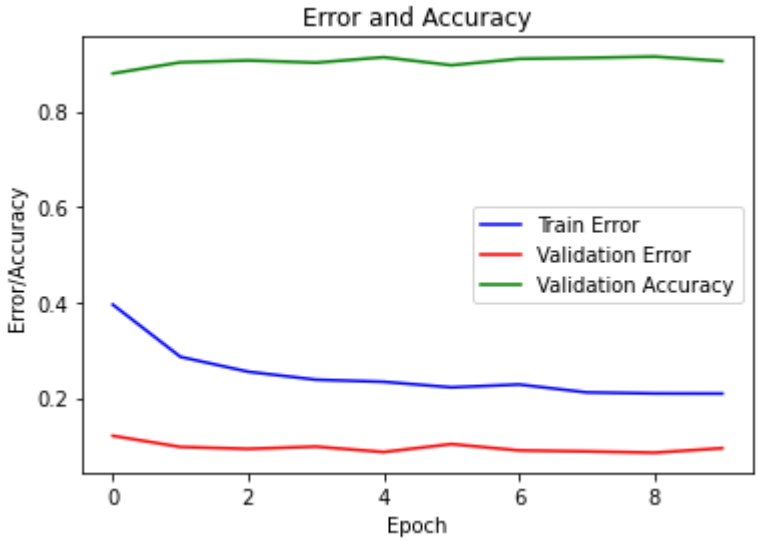
2.1. Task 02 Fine-tune the CNN and FC layers of the network:

2.1.1. VGG-16 Entire

2.1.1.1. Environment Setup:

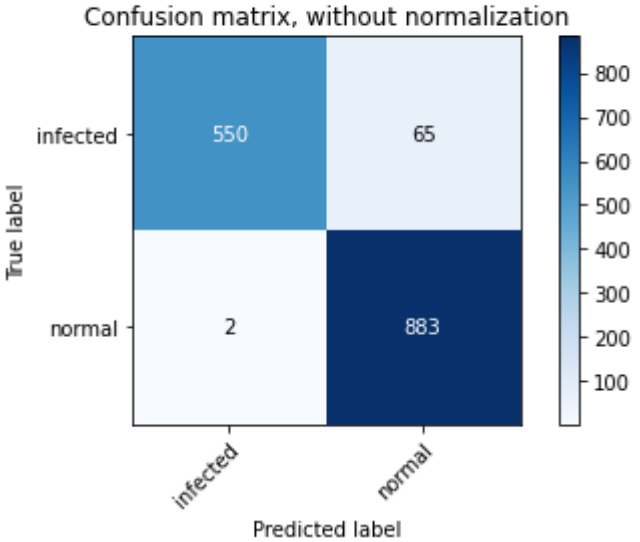
Epochs	=	10
Learning Rate	=	0.001
Feature Layers	=	unfreeze
Fine-tune layers	=	All feature layers and FC layers are Fine-tune

2.1.1.2. Loss and Accuracy Curve:



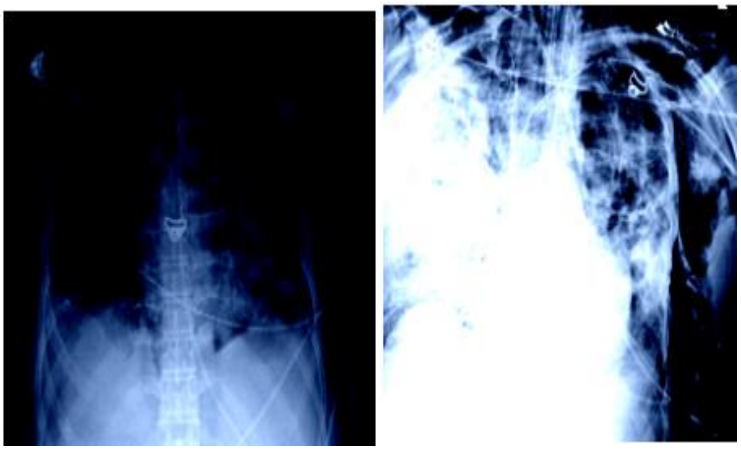
2.1.1.3. Confusion Matrices:

```
Confusion matrix, without normalization
[[550  65]
 [  2 883]]
```

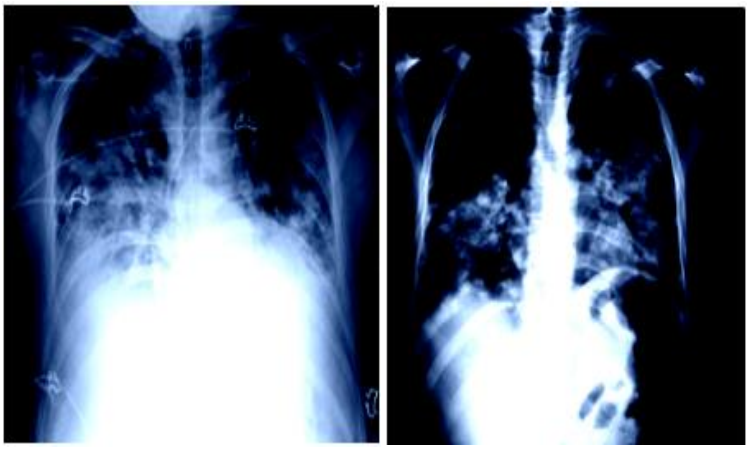


2.1.1.4. Best and Worst Classified Images:

2.1.1.4.1. Best Classified Infected Images:

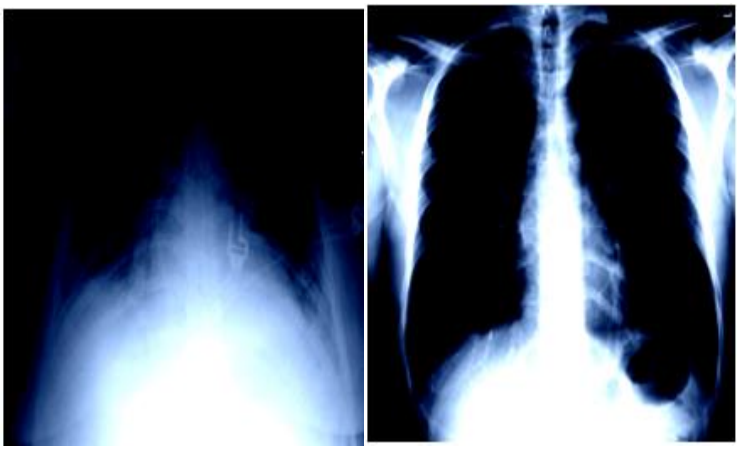


2.1.1.4.2. Best Classified Normal Images:



2.1.1.4.3. Worst Classified Infected Images:

2.1.1.4.4. Worst Classified Normal Images:



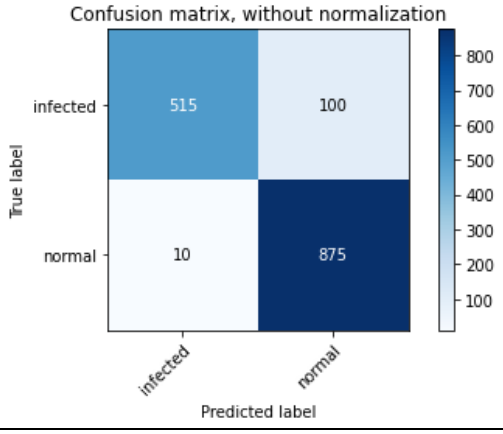
2.1.1.5. Final Accuracy and F1 Score:

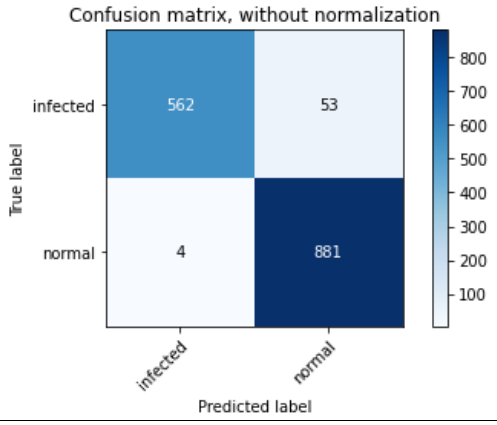
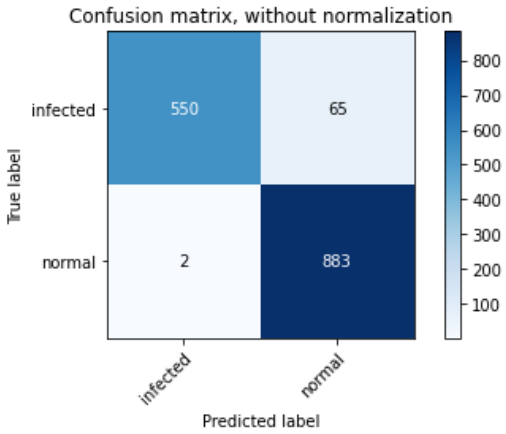
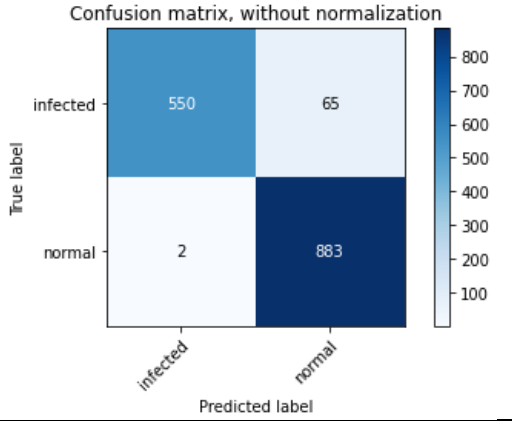
The final test accuracy for this network is ***95.53%*** and F1 score is ***0.96.***

2.1.1.6. Experiments:

Model: VGG16

Table 1 VGG16 Experiments

Experiment	Accuracy	F1 Score	Confusion Matrix
Only FC layers Fine-tune	92.66%	0.94	Confusion matrix, without normalization [[515 100] [10 875]] 

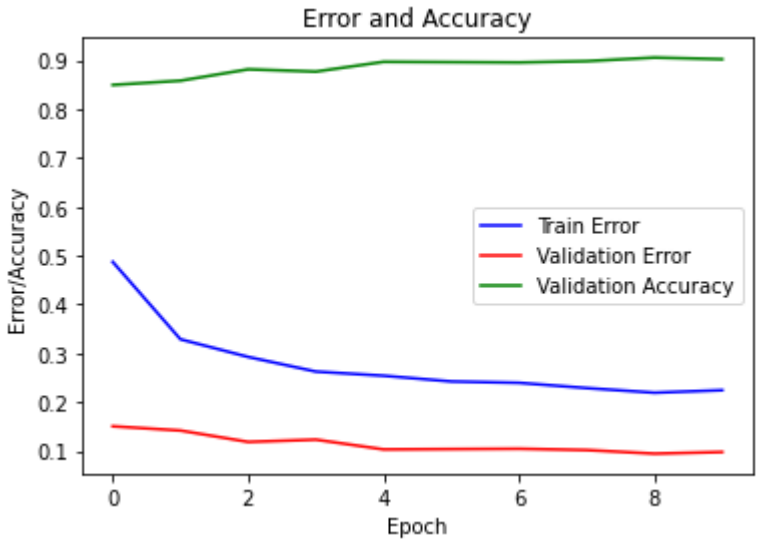
One Conv2d (feature layer 10) and FC layers Fine-tune	96.53%	0.97	Confusion matrix, without normalization [[562 53] [4 881]] 
Few Conv2d layers (0, 5, 10, 12, 19, 21, 26, 28) and FC layers Fine-tune	95.53	0.96	Confusion matrix, without normalization [[550 65] [2 883]] 
Entire Model Fine-tune (Both Feature and FC layers)	95.53%	0.96	Confusion matrix, without normalization [[550 65] [2 883]] 

2.1.2. ResNet18 Entire:

2.1.2.1. Environment Setup:

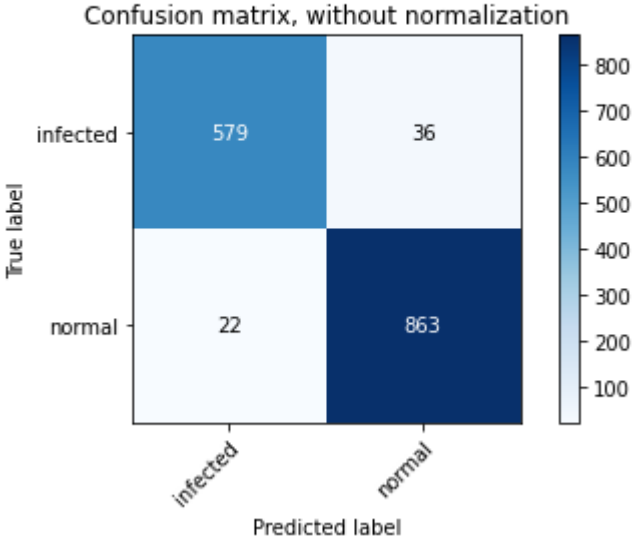
Epochs	=	10
Learning Rate	=	0.001
Feature Layers	=	unfreeze
Fine-tune layers	=	All features and FC layers

2.1.2.2. Loss and Accuracy Curve:



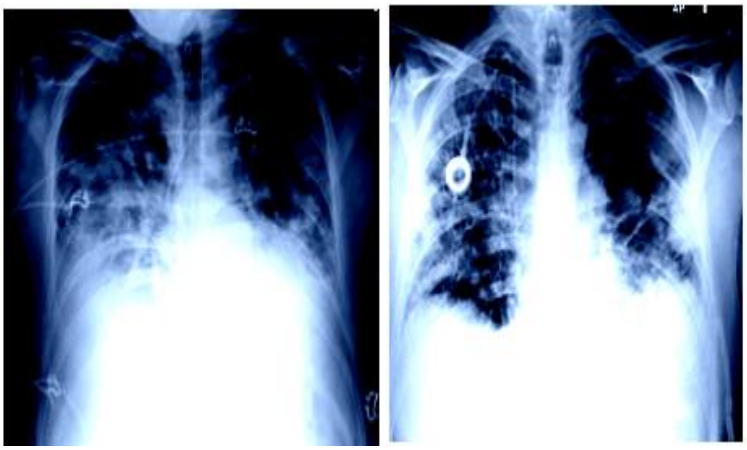
2.1.2.3. Confusion Matrix:

```
Confusion matrix, without normalization
[[579  36]
 [ 22 863]]
```

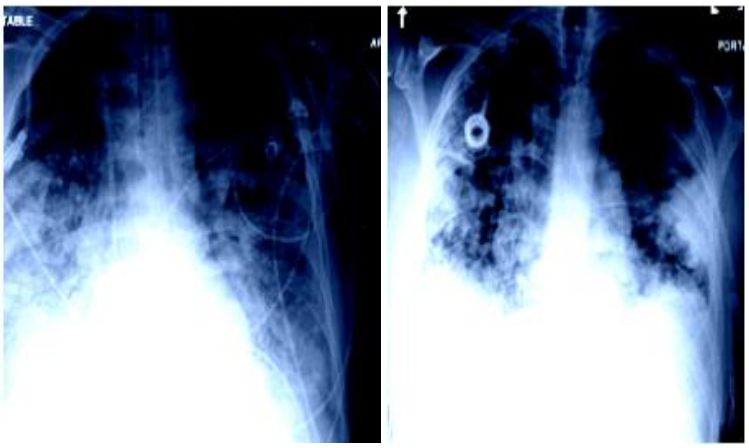


2.1.2.4. Best and Worst Classified Images:

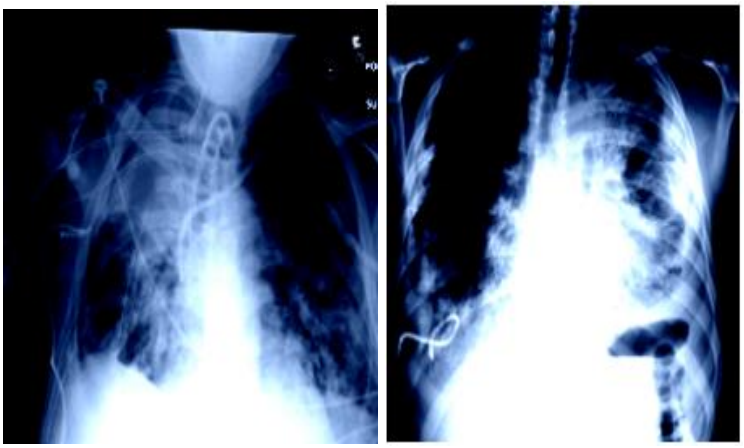
2.1.2.4.1. Best Classified Infected Images:



2.1.2.4.2. Best Classified Normal Images:



2.1.2.4.3. Worst Classified Infected Images:



2.1.2.4.4. Worst Classified Normal Images:

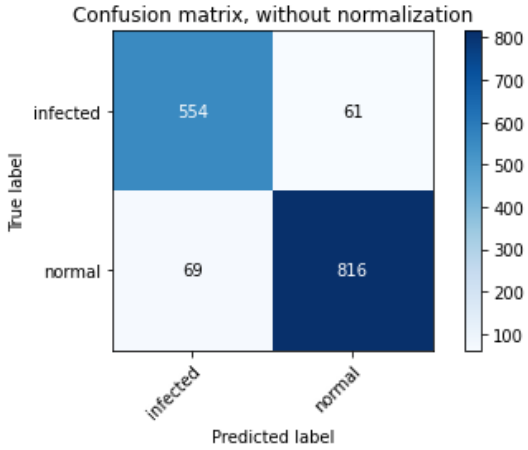
2.1.2.5. Final Accuracy and F1 Score

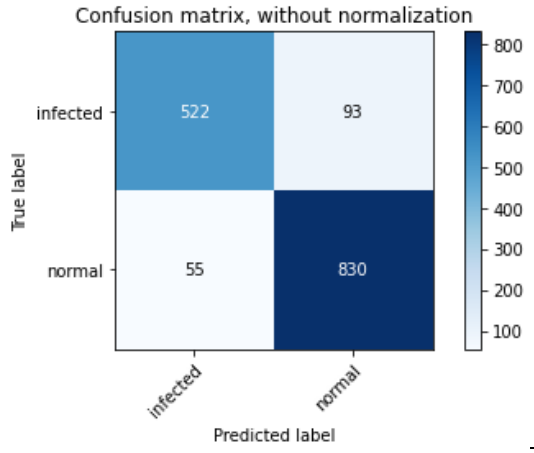
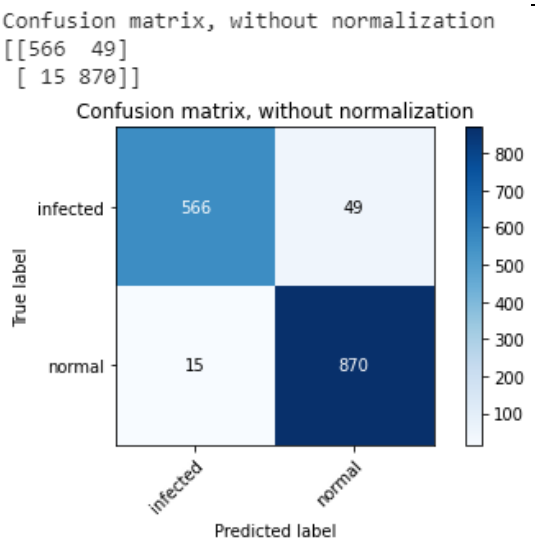
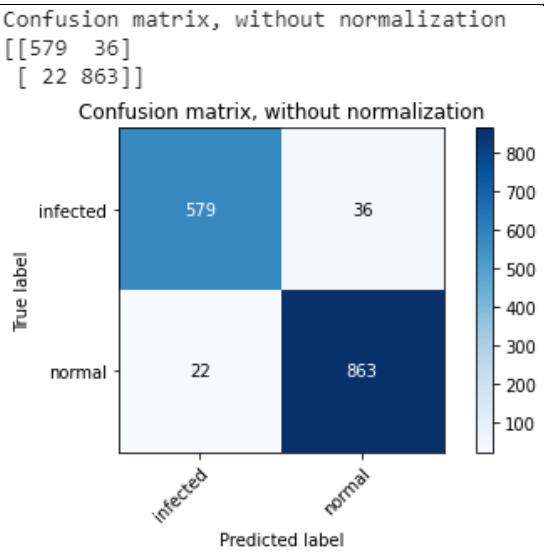
The final test accuracy for this network is ***96.13%*** and F1 score is ***0.97***

2.1.2.6. Experiments:

Model: ResNet18

Table 2 ResNet18 Experiments

Experiment	Accuracy	F1 Score	Confusion Matrix
Only FC layers Fine-tune	91.3%	0.93	Confusion matrix, without normalization [[554 61] [69 816]] 

One Conv2d (feature layer 10) and FC layers Fine-tune	90.6%	0.92	Confusion matrix, without normalization [[522 93] [55 830]] 
Few Conv2d layers (conv1, layer1, layer3) and FC layers Fine-tune	95.53	0.96	Confusion matrix, without normalization [[566 49] [15 870]] 
Entire Model Fine-tune (Both Feature and FC layers)	96.13%	0.97	Confusion matrix, without normalization [[579 36] [22 863]] 

2.1.3. Analysis:

In VGG16 10th layer of model which is covn2d layer is unfreeze this layer fine-tune with respect to this dataset due to which it has highest accuracy (96.53%), it is higher than all fine-tune model. From this I conclude that we can check every possibility of freeze and unfreeze it help us to find better accuracy. But in ResNet18 entire fine-tune model gives higher accuracy which is 96.13%. The Main reason that more layer fine-tune with dataset it gives more accuracy as you can see in Table1 and Table2 above.