DETECTING PNEUMPNIA/COVID19 IN CHEST X-RAY DATASET USING CNN

-Submitted by-

SHANKARANARAYAN N

203IT001

I SEM - M.Tech(Research) IT

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA, SURATHKAL

OCT 2020-JAN2021

ABSTRACT :

Pneumonia and Covid-19 are both life-threatening infectious diseases affecting the lungs of the human body. A recent research on covid-19 has revealed that even after a patient's complete recovery from COVID-19 he/she can suffer from life long side effects. One of which is the risk of getting viral/bacterial pneumonia. Tests to predict covid-19 are a bit time consuming and is not accessible to everyone. More over as the symptoms of Covid-19 and Pneumonia are quiet similar there is a huge chance that the test resources are wasted on false alarms assuming Pneumonia as Covid-19.In this scenario, detection with the help of X-ray may help us identify the infection and take necessary measures and to reduce the number of unnecessary false tests. In this project, we will be applying the convolutional neural network architectures such as RESNET-18 and DENSENET which overcomes the disadvantages of traditional convolutional neutral networks and compare them to identify the best architecture to predict covid-19/pneumonia.

Keywords : Pneumonia, Covid-19, Convolutional Neural Networks, RESNET18, DENSENET

CONTENTS

1.INTRODUCTION

Coronavirus disease 2019 (COVID-19) is a contagious disease caused by severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). Pneumonia is an inflammatory condition of the lung primarily. Both of them are lung diseases and exhibit similar symptoms such as productive or dry cough, chest pain, fever and difficulty breathing. Because of the similarities in the symptoms, people with pneumonia symptoms often tend to get confused and try to get a COVID-19 test. But, in the current scenario considering the less availability of the testing kits adopting X-ray classification to distinguish between Pneumonia and Codiv-19 can save a lot of resources.

In the project we will be performing classification between Pneumonia, covid-19 and normal X-ray images with the help of convolutional neural network. We know that the traditional convolutional neural network models suffers for issues such as vanishing gradient descent. In simple words, medical image classification very much depends upon on the details on the images. But the traditional convolutional model tends to loose these details proportionally to the number of layers. So, here in this project we will be using CNN architectures such as RESNET-18 and DENSENET that does not suffer from these issues. And at the end we will be comparing these to models with respect to accuracy and training iteration.

The dataset used in the project is COVID-19 RADIOGRAPHY DATABASE (Winner of the COVID-19 Dataset Award by Kaggle Community). We will be looking into the details of this in the succeeding sections.

2.LITRATURE SURVEY

2.1 CURRENT SITUATION OF COVID19 RESEARCH

Most nations had to take measures to react to the sudden and rapid outbreak of COVID-19 within a relatively short period of time. Despite of the current technology the research on COVID 19 has suffered and delayed a lot due to the less availability of data on covid19. All the research happening on COVID19 have access only to a limited amount of data. COVID-19 is a respiratory virus which mainly affects our lung. So medical imaging such as XRAY can help us identify COVID. Other data such as body temperature, body pain can help us detect COVID, But after conventional testing, medical imaging is the best way to detect COVID.

2.2 RADIOGRAPHY AND COVID19 RESEARCH

Several research work exist in the literature that exploited various deep learning techniques on X-ray data to demonstrate reasonable performance . In one of the model, referred to as DarkCovidNet, for early detection of COVID-19 was proposed which utilized 17 convolutional layers to perform binary and multi-class classification involving normal, COVID, and pneumonia cases. Several other papers applied deep learning models on CT scan images to detect and monitor COVID-19 features in the radiograph data . Some employed implemented the state of the art CNN architectures such as AlexNet, ResNet-18, ResNet-50, ResNet-101, SqueezeNet, VGG-16, VGG-19, MobileNet-V2, GoogleNet, and XceptionCT to differentiate between COVID-19 and non-COVID-19 cases. Their experiments showed that deep learning could be considered as a feasible technique for identifying COVID-19 from radiograph images.

3. METHODOLOGY AND RESULT:

3.1 TECHNOLOGY STACK : The entire project is done in **GoogleColab** environment. And the dataset for training is stored and accessed from **Google Drive**. And the CNN models are constructed using **Keras**, **Tensorflow** and **Pytorch**.

3.2 DATASET : COVID-19 RADIOGRAPHY DATABASE (Winner of the COVID-19 Dataset Award by Kaggle Community)

The dataset used in the project is taken from Kaggle. The dataset contain 3 different sets of chest x-ray images(covid19-1143, pneumonia-1345, normal-1341). We will be splitting the data into training and test sets for training and testing purposes.

All the data that is used for training and testing purposes are stored in Google Drive and is accessed with the help of an authentication key.

3.3 STEPS INVOLVED :

STEP 1 ACCESSING THE DATASET

- AS THE DATASET REQUIRED FOR TRAINING IS STORED IN GOOGLE DRIVE THE FIRST STEP IS TO ESTABLISH CONNECTION TO THE DRIVE AND MOUNT IT.

STEP 2 PREPARING TRAINING AND TEST SETS

- IN THIS STEP WE WILL BE DRIVING THE DATA INTO TRAINING AND TEST SETS

STEP 3 IMAGE TRANSFORMATION

- IN THIS STEP WE PERFORM DATA AUGUMENTATION BY RESCALING THE IMAGE

STEP 4 CREATING MODEL

- IN THIS STEP WE CREATE THE CNN MODEL

STEP 5 TRAINING MODEL

- TRANING IS DONE HERE WITH RESPECT TO THE TEST AND TRAIN DATA

STEP 6 PLOTTING RESULT

- THIS IS THE FINAL STEP WHERE WE PLOT THE RESULT

3.4 MODELS USED :

The project itself is divided into 3 Parts based on the type of CNN architecture in use.

Part 1 : Classification through Naïve approach

Part 2 : Classification through RESNET18

Part 3 : Classification through DENSENET

4.EXECUTION AND RESULT

4.1.1 PART 1 : CLASSIFICATION THROUGH NAÏVE APPROACH

In this part we are using a naïve approach to classify the images. (i.e) We are not using any standard architecture. We are constructing a convolutional neural network with the help of Keras and Tensorflow. The below given is the summary of the created convolutional model.

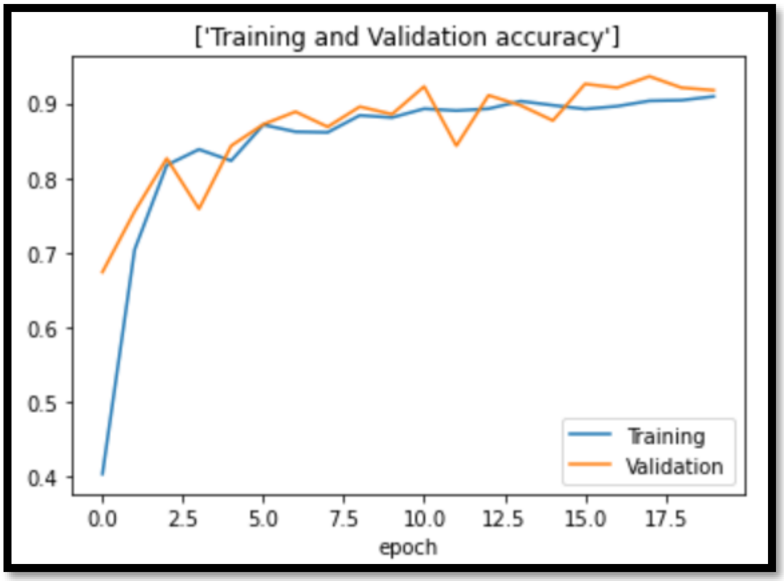
```
Model: "sequential_16"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_58 (Conv2D)           (None, 148, 148, 32)      896
_________________________________________________________________
max_pooling2d_57 (MaxPooling (None, 74, 74, 32)        0
_________________________________________________________________
conv2d_59 (Conv2D)           (None, 72, 72, 32)        9248
_________________________________________________________________
max_pooling2d_58 (MaxPooling (None, 36, 36, 32)        0
_________________________________________________________________
conv2d_60 (Conv2D)           (None, 34, 34, 64)        18496
_________________________________________________________________
max_pooling2d_59 (MaxPooling (None, 17, 17, 64)        0
_________________________________________________________________
conv2d_61 (Conv2D)           (None, 15, 15, 64)        36928
_________________________________________________________________
max_pooling2d_60 (MaxPooling (None, 7, 7, 64)          0
_________________________________________________________________
conv2d_62 (Conv2D)           (None, 5, 5, 64)          36928
_________________________________________________________________
max_pooling2d_61 (MaxPooling (None, 2, 2, 64)          0
_________________________________________________________________
flatten_15 (Flatten)         (None, 256)               0
_________________________________________________________________
dense_30 (Dense)             (None, 128)               32896
_________________________________________________________________
dense_31 (Dense)             (None, 3)                 387
=================================================================
Total params: 135,779
Trainable params: 135,779
Non-trainable params: 0
_________________________________________________________________
```

We perform the training with the following settings,

- Learning Rate : 0.01
- Loss Function Calculation : Categorical Crossentropy
- Batch size = 16
- Training Iteration(EPOCH) = 15

4.1.2 PART 1 : CLASSIFICATION THROUGH NAÏVE APPROACH : RESULT

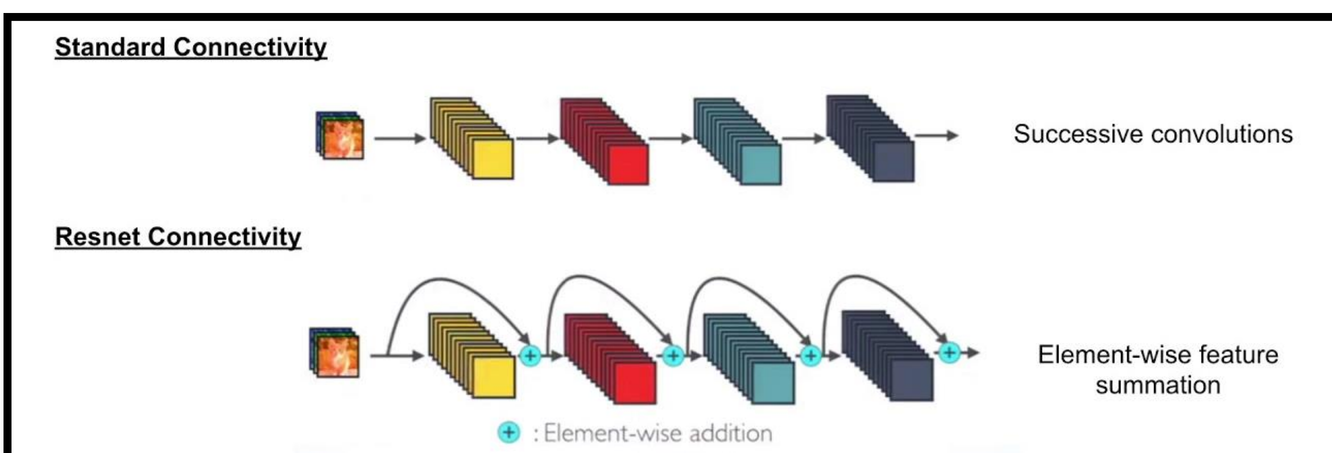
We obtain the following result as a result of the naïve approach and with 15 iterations the highest accuracy achieved is 54%



THE GRAPH DISPLAYS THE TRAINING AND VALIDATION ACCURACY OF THE NAIVE MODEL WITH RESPECT TO THE ITERATION.

4.2.1 PART 2 : CLASSIFICATION THROUGH RESNET18

RESNET18 overcomes most of the issues faced by the traditional convolutional neural networks. To be precise the main reason that we prefer RESNET18 is that, traditional CNN models tend to loose the minute details in the image after certain layer. But RESTNET18 overcomes this issue by contaminating values from the precious layer.



MATHEMATICAL EXPLANATION

$$z^{[l+1]} = W^{[l+1]}\, a^{[l]} + b^{[l+1]} \qquad a^{[l+1]} = g(z^{[l+1]})$$

$$z^{[l+2]} = W^{[l+2]} a^{[l+1]} + b^{[l+2]} \qquad a^{[L+2]} = g\left(z^{[L+2]}\right)$$

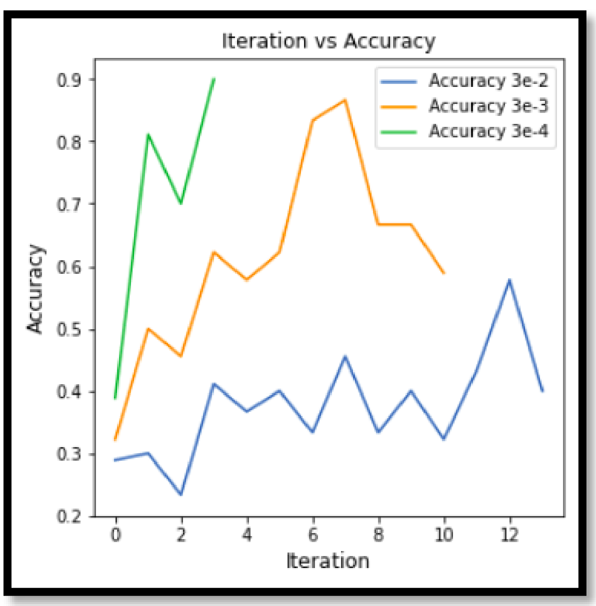
$$\boxed{a^{[L+2]} = g\left(z^{[L+2]} + a^{L}\right)}$$

Initially,We perform the training with the following settings,

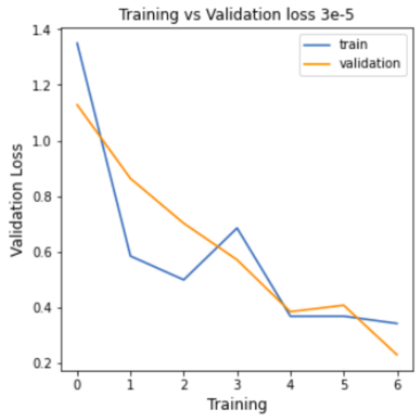
- Learning Rate : 3x(10^-2) or 3e-2 , 3e-3, 3e-4
- Loss Function Calculation : Categorical Crossentropy
- Batch size = 6
- Training Iteration(EPOCH) = Till accuracy is more than 90%

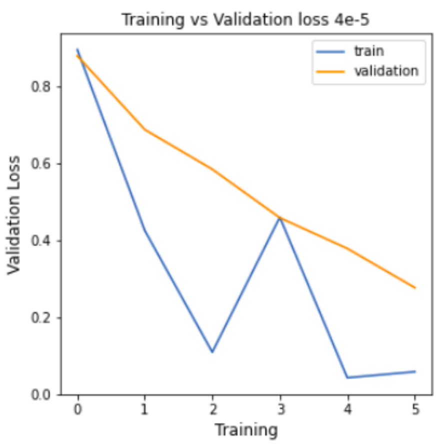
We are doing this inorder to identify an appropriate learning rate

4.2.2 IDENTIFYING APPROPRIATE TRAINING RATE : RESNET



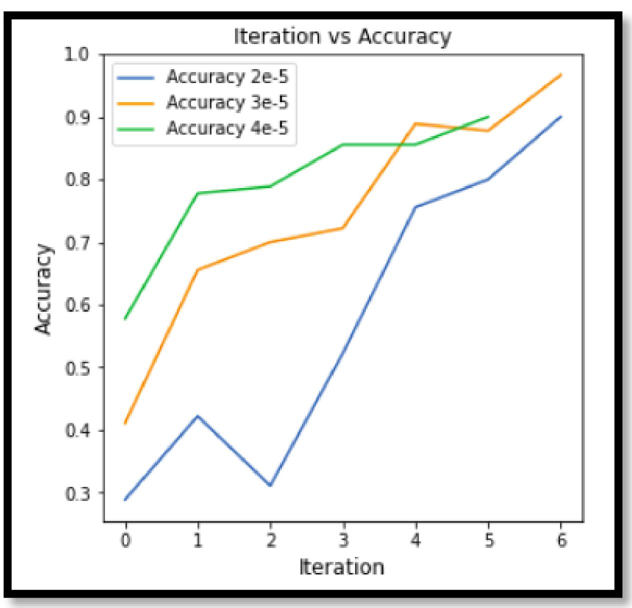
Here, we have plotted the accuracy vs Iteration graph for the above discussed learning rates. Out of the discussed parameters accuracy rate [3e-4] attains an accuracy of 90+. But as the graph displays the accuracy is increasing in a rapid manner. So, we are deciding to go with a learning rate of Xe-5 which might give us a steady progress.





Training and Validation graph for the figure learning rates discusses here.

4.2.3 ACCURACY VS ITERATION GRAPH : RESNET



Here, we train our model with a learning rate of [2e-5,3e-5,4e-5]

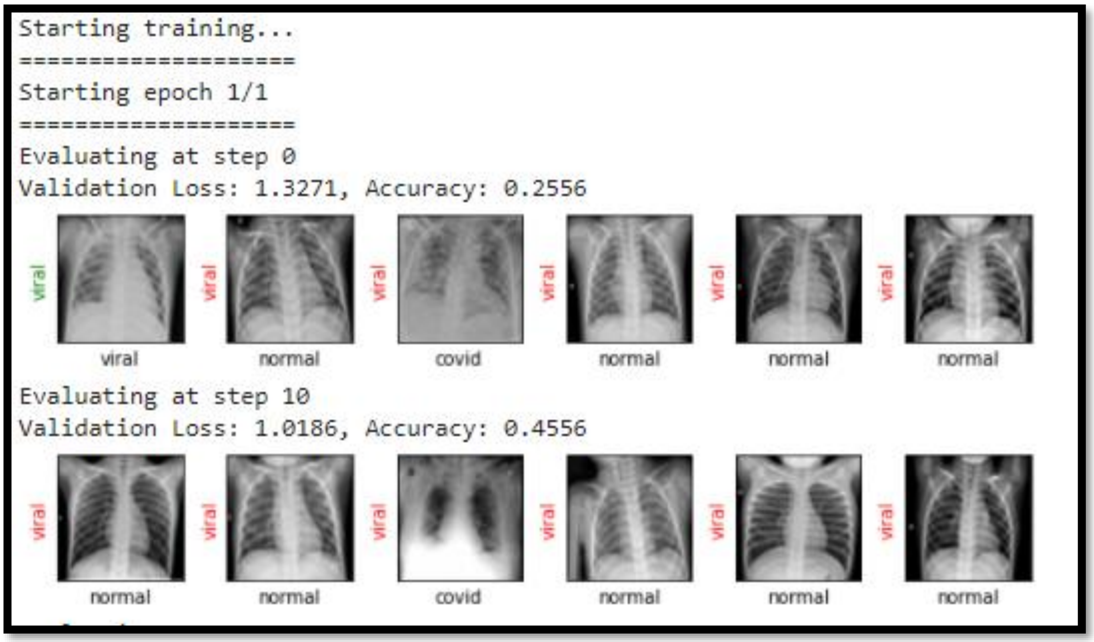
We iterate till we reach a satisfactory accuracy of more than 90%. We can notice that all of them have a stable increase in accuracy.

X-axis : 1unit = 10 iteration

4.2.4 PART 2 : CLASSIFICATION THROUGH RESTNET18 - RESULT

PLEASE NOTE THAT THE X-AXIS DENOTES THE ACTUAL VALUE AND THE Y-AXIS DENOTES THE PREDICTED VALUE

PREDICTION BEING TRUE IS IN GREEN, PREDICTION BEING FALSE IS IN RED.

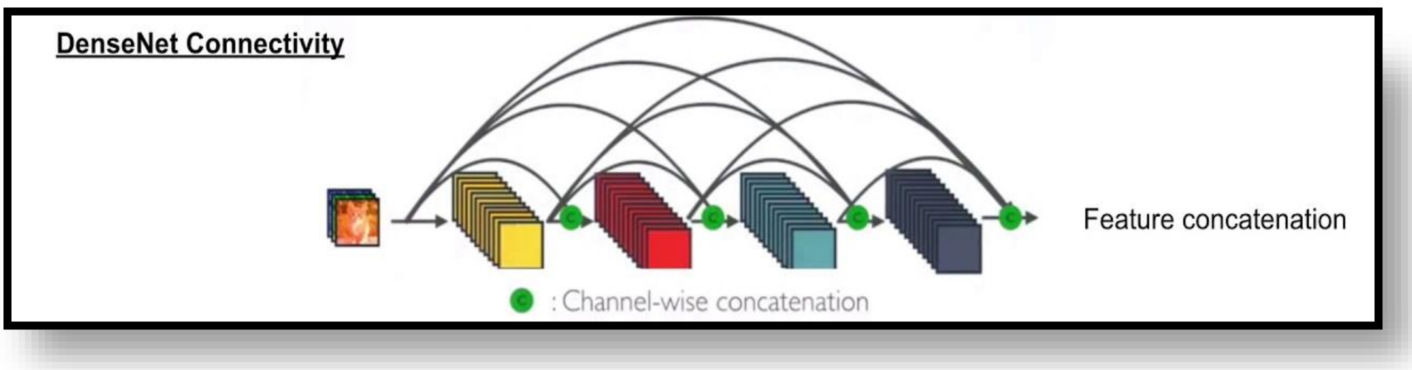


The training with 25% and the model is trained till it reaches an accuracy more tha 90%

The model training finally terminates when the accuracy reaches 93% at the 50th iteration.

4.3.1 PART 3 : CLASSIFICATION THROUGH DENSENET

In Standard ConvNet, input image goes through multiple convolution and obtain high-level features. In ResNet, identity mapping is proposed to promote the gradient propagation. Element-wise addition is used. It can be viewed as algorithms with a state passed from one ResNet module to another one. In DenseNet, each layer obtains additional inputs from all preceding layers and passes on its own feature-maps to all subsequent layers. Concatenation is used. Each layer is receiving a "collective knowledge" from all preceding layers. Since each layer receives feature maps from all preceding layers, network can be thinner and compact, i.e. number of channels can be fewer. The growth rate k is the additional number of channels for each layer. So, it have higher computational efficiency and memory efficiency. The following figure shows the concept of concatenation during forward propagation.



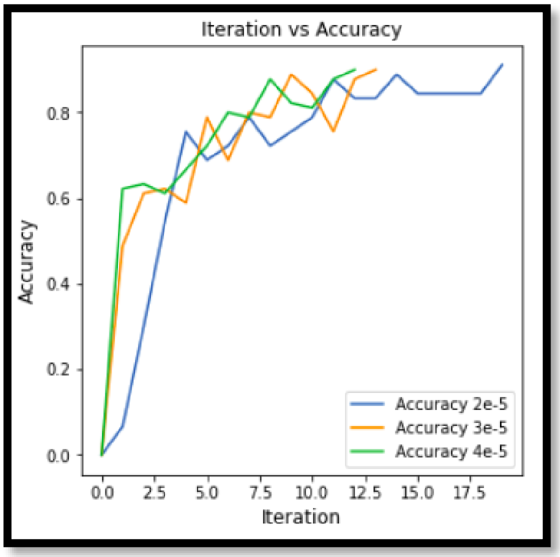
We perform the training with the following settings,

- Learning Rate : 2e-5, 3x(10^-5) or 3e-5, 4e-4 [In accordance with previous testing in RESNET]
- Loss Function Calculation : Categorical Crossentropy
- Batch size = 6
- Training Iteration(EPOCH) = Till accuracy is more than 90%

4.3.2 TRAINING VS VALIDATION GRAPH : DENSENET

The above figure shows us the training and validation loss graph for different.

4.3.4 ACCURACY VS ITERATION GRAPH : DENSENET

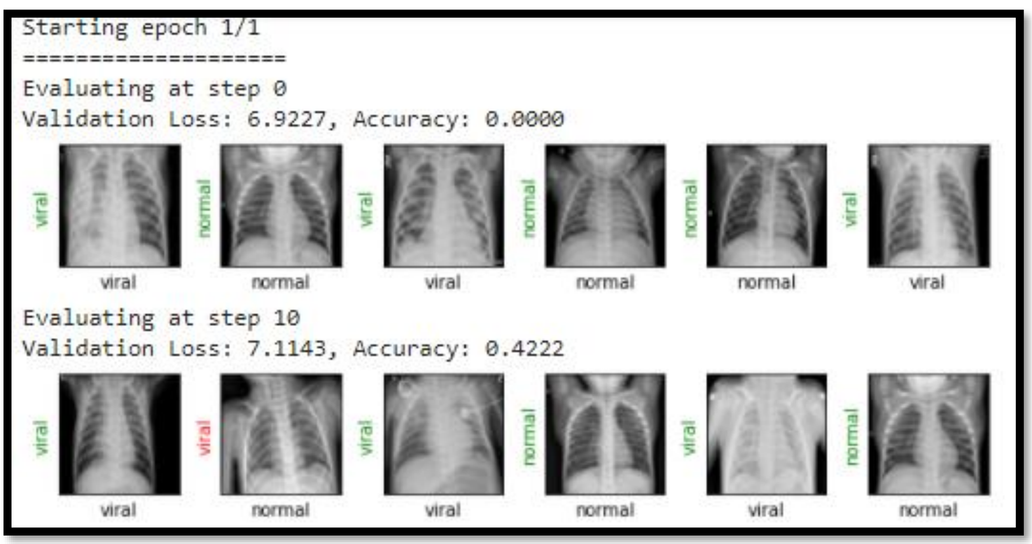


Here it can be noted that densenet architecture takes more time to perform the training. This is because of the more number of layers and features when compared to the restnet architecture.

Also note that densenet maintains more of original information.

4.3.5 PART 3 : CLASSIFICATION THROUGH DENSENET - RESULT

PLEASE NOTE THAT THE X-AXIS DENOTES THE ACTUAL VALUE AND THE Y-AXIS DENOTES THE PREDICTED VALUE. PREDICTION BEING TRUE IS IN GREEN, PREDICTION BEING FALSE IS IN RED.

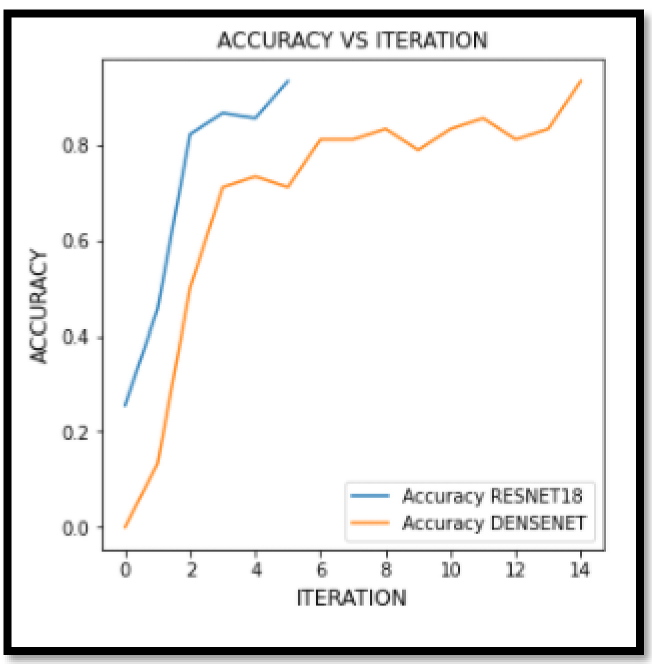


The training with 0% accuracy and the model is trained till it reaches an accuracy more tha 90%

The model training finally terminates when the accuracy reaches 93% at the 120th iteration.

5. RESULT, CONCLUSION AND FUTURE WORK

5.1 RESULT COMPARISON



THIS GRAPH ILLUSTRATES THE NUMBER OF ITERATIONS TAKEN BY BOTH THE RESNET MODEL AND DENSENET MODEL TO REACH AN ACCURACY OF MORE THAN 90%

X-AXIS : 1UNIT = 10 ITERATION

From the above graph we can conclude that the RESNET MODEL reached the required accuracy fare before the DENSENET MODEL this difference in training iteration could be possibly because of the nubmer of layes that are involved. The RESNET model that we use only involved 18 Layers and 512 Features but the DENSENET MODEL that we use 131 layers and 2208 features

5.2 CONCLUSION

As discussed before, classification related to medical data such as X-Ray data classification and MRI data classification should retain even the minute details. As it is well known that traditional Convolutional Neural Network models fails to retain information as the number of layers increase[Information vanishes as we go deeper]. We adopted RESNET AND DENSENET. But even in these models we might face some issues. IE- RESNET may reach a platue after certain number of training or it may also loose the minute information as number of layers increase even further. So as conclution we recommend DENSENET ARCHITECTURE as it retains the most of the original information.

5.3 FUTURE WORK

COVID 19 has an incubation periord of 15 days. IE- The infection can be detected only after 15 days after initial contact till which point the affected person could be asymptomatic. This could further spread the indection to a lot more people. I propose the following model with which we could be able to detect covid during the incubation periord itself.

PROPOSAL : To create a dataset consisting of xray images and health data from the time of initial contact to the time of full development. Even with a small initial dataset. We could have the aid of deep generative models to creat emore such dataset. Now implementing rnn in this sequense data might help us predict covid in the incubation periord itself.

6. REFERENCE :

1. **Can AI Help in Screening Viral and COVID-19 Pneumonia?**
   [IEEE ACCESS]
   Digital Object Identifier 10.1109/ACCESS.2020.3010287

2. **DL-CRC: Deep Learning-Based Chest Radiograph Classification for COVID-19 Detection: A Novel Approach**
   [IEEE ACCESS]
   Digital Object Identifier 10.1109/ACCESS.2020.3025010

3. **Diagnosis of Pneumonia from Chest X-Ray Images Using Deep Learning**
   [IEEE EXPLORE]
   https://ieeexplore.ieee.org/document/8741582

4. **Covid Prediction from X-ray Images**
   [IEEE EXPLORE]
   https://ieeexplore.ieee.org/document/9276795