COVID-19 DETECTION USING CONVOLUTIONAL NEURAL NETWORK

**PROJECT REPORT**

**Submitted by:**

# SHUBHI GIRI (180050101055)

# GAURAV JOSHI (180050101021)

# PRATEEK BHAISORA (180050101039)

# KUNAL JOSHI (180050101028)

# HARSHIT SHARMA (180050101026)

****

**Under the Mentorship of:**

# MR. NEERAJ BISHT

**(Assistant Professor)**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**BIRLA INSTITUTE OF APPLIED SCIENCES**

**BHIMTAL, UTTARAKHAND - 263136**

**MAY, 2022**

**DECLARATION**

We the undersigned solemnly declare that the project report “Covid-19 Detection using Convolutional Neural Network” is based on our own work carried out during the course of our study under the supervision of **Mr. Neeraj Bisht**.

We assert the statements made and conclusions drawn are an outcome of our research work. I further certify that:

1. The work contained in the report is original and has been done by us under the general supervision of our supervisor.
2. The work has not been submitted to any other Institution for any degree/diploma/ certificate in this university or any other University of India or abroad.
3. We have followed the guidelines provided by the university in writing the report.
4. Whenever we have used materials (data, theoretical analysis, and text) from other sources, we have given due credit to them in the text of the report and giving their details in the references.

BIRLA INSTITUTE OF APPLIED SCIENCES **Name Roll No**

# BHIMTAL, UTTARAKHAND – 263136 SHUBHI GIRI (180050101055)

# GAURAV JOSHI (180050101021)

# Date: PRATEEK BHAISORA (180050101039)

# KUNAL JOSHI (180050101028)

# HARSHIT SHARMA (180050101026)

# 

# 

# ACKNOWLEDGEMENT

We would like to express our deep sense of gratitude to our Mentor **Mr. Neeraj Bisht** - **Assistant Professor**, Department of **Computer Science & Engineering**, Birla Institute of Applied Sciences, Bhimtal, Uttarakhand for his guidance, keen interest, constant encouragement and helpful. His advice was invaluable even when the going got tough during the course of the study embodied in this seminar work.

Our deepest gratitude also goes for our respected Director Sir, Mr. Abhay Sharma and HOD, Dr. Sandesh Tripathi for their valuable advices, encouragement and all time support.

We would like to thank all faculties and staff-members of the Computer Science Engineering & Department, at BIAS Bhimtal for their help and suggestions for betterment of the seminar work and its presentation. We also extend our sincere regards to Uttarakhand Technical University (UTU) Dehradun, Uttarakhand for providing infrastructure development support.

We would like to thank our friends and colleagues for critically reading the manuscript and providing information at times. They are Sherya Joshi,

Harshit Pathak and Abhishek Loshali.

We are also highly grateful to our parents for their endless support.

Date: **Name Roll No.**

# SHUBHI GIRI (180050101055)

# GAURAV JOSHI (180050101021)

# Place: Bhimtal PRATEEK BHAISORA (180050101039)

# KUNAL JOSHI (180050101028)

# HARSHIT SHARMA (180050101026)

**Checked & Verified by:**

# ABSTRACT

COVID-19 global pandemic affects health care and lifestyle worldwide, and its early detection is critical to control cases’ spreading. As it rapidly spreads across the planet, scientists and public-health experts are racing to slow down the spreading and trying to find methodologies to detect it. The actual leader diagnosis test is the Reverse transcription Polymerase chain reaction (RT-PCR), result times and cost of these tests are high, so other fast and accessible diagnostic tools are needed. Thus, we developed a model that automatically detect COVID and non-COVID X-rays. These days Deep Learning algorithms showing the foremost results in Disease classification. Also, features learned by pre-trained Convolution Neural Networks (CNN) models on large-scale datasets are much useful in image classification tasks. We train and test our model to analyze the images as COVID or normal. We analytically determine the optimal CNN model for the purpose.

This project approach uses Convolutional Neural Network to process these images and classify them as positive or negative for COVID-19. The proposed system involves a preprocessing stage with lung segmentation, removing the surroundings which does not offer relevant information for the task and may produce biased results; after this initial stage comes the classification model trained under the transfer learning scheme; and finally, results analysis and interpretation via heat maps visualization.

Due to the non-availability of sufficient-size and good-quality chest X-ray image dataset, an effective and accurate CNN classification was a challenge. To deal with these complexities such as the availability of a very-small-sized and imbalanced dataset with image-quality issues, the dataset has been preprocessed in different phases using different techniques to achieve an effective training dataset for the proposed CNN model to attain its best performance. The best models achieved a detection accuracy of COVID-19 around 98%.

**TABLE OF CONTENTS**

[1) INTRODUCTION 1](#_Toc104462511)

[2) LITERATURE REVIEW 5](#_Toc104462512)

[2.1) COVID-19 Diagnosis Using Deep Learning 5](#_Toc104462513)

[2.2) X-Ray Diagnosis Using Deep Learning 6](#_Toc104462514)

[3) TECHNOLOGY AND METHODS 11](#_Toc104462515)

[3.1) Dataset 11](#_Toc104462516)

[3.3) Training model 13](#_Toc104462517)

[3.3.a) Convolutional Neural Network (CNNs): 13](#_Toc104462518)

[3.4) MODEL ARCHITECTURE:- 15](#_Toc104462520)

[3.5.b) Keras 17](#_Toc104462521)

[3.5.c) NumPy 17](#_Toc104462522)

[3.5.d) OS Module 18](#_Toc104462523)

[3.5.e) Open-CV 18](#_Toc104462524)

[3.5.f) Pandas 18](#_Toc104462525)

[3.6) VISUALIZATION:- 20](#_Toc104462526)

[3.6.b) Seaborn 20](#_Toc104462527)

[3.6.c) HTML 20](#_Toc104462528)

[3.6.d) CSS 21](#_Toc104462529)

[3.6.e) jQuery 21](#_Toc104462530)

[3.6.f) Flask Web Framework 22](#_Toc104462531)

[3.6.g) Pillow 23](#_Toc104462532)

[3.7) DATA PREPARATION 24](#_Toc104462533)

[3.9) WEB:- 26](#_Toc104462534)

[3.9.b) Index.html (HTML+ CSS + jQuery) 27](#_Toc104462535)

[5) CONCLUSION 34](#_Toc104462536)

[6) REFERENCES 35](#_Toc104462537)

**TABLE OF FIGURES**

[Figure 1.1: X-Ray images of Covid Positive 12](#_Toc104462882)

[Figure 2: A sample Covid-19 Dataset from Github 13](#_Toc104462883)

[Figure 3: Layers in CNN 15](file:///C:\Users\Dell\Desktop\project_report.docx#_Toc104462884)

[Figure 4: Table showing Total params, Total Trainable and Non-trainable params 16](#_Toc104462885)

[Figure 5: Test loss, test accuracy, train loss and train accuracy obtained after testing 30](file:///C:\Users\Dell\Desktop\project_report.docx#_Toc104462886)

# INTRODUCTION

Coronavirus illness is a disease that comes from Severe Acute Respiratory Syndrome (SARS) and Middle East Respiratory Syndrome (MERS). A novel coronavirus, COVID-19, is the infection caused by SARS-CoV-2 (Zhang, 2020). In December 2019, the first COVID-19 cases were reported in Wuhan city, Hubei province, China (Xu et al.,2020). World Health Organization (WHO) declared COVID-19 a pandemic (Ducharme, 2020) on March 11 2021, up to January of 2022 there are 32 Crore reported cases around the world, which have caused 55.2 Lakh deaths worldwide. These diseases cause respiratory problems that can be treated without specialized medicine or equipment. Still, underlying medical issues such as diabetes, cancer, cardiovascular and respiratory illnesses can make this sickness worse (World Health Organization, 2021). The most serious illness caused by COVID-19 is related to the lungs such as pneumonia. The symptoms of the disease can vary and include dyspnea (shortness of breath), high fever, runny nose, and cough. These cases can most commonly be diagnosed using chest X-ray imaging analysis for the abnormalities.

Reverse transcription Polymerase chain reaction (RT-PCR), gene sequencing for respiratory or blood samples are now the main methods for COVID-19 detection. Other studies show that COVID-19 has similar pathologies presented in pneumonic illness, leaving chest pathologies visible in medical images. Research shows RT-PCR correlation with Chest CT, while others study its correlation with X-ray chest images. Typical opacities or attenuation are the most common finding in these images, with ground-glass opacity in around 57% of cases. Even though expert radiologists can identify the visual patterns found in these images, considering monetary resources at low-level medical institutions and the ongoing increase of cases, this diagnostic process is quite impractical. Recent research in Artificial Intelligence (AI), especially in Deep Learning approaches, shows how these techniques applied to medical images performed well.

X-radiation or X-ray is an electromagnetic form of penetrating radiation. These radiations are passed through the desired human body parts to create images of internal details of the body part. The X-ray image is a representation of the internal body parts in black and white shades.

X-ray is one of the oldest and commonly used medical diagnosis tests. Chest X-ray is used to diagnose the chest-related diseases like pneumonia and other lung diseases, as it provides the image of the thoracic cavity, consisting of the chest and spine bones along with the soft organs including the lungs, blood vessels, and airways.

The X-ray imaging technique provides numerous advantages as an alternative diagnosis procedure for COVID-19 over other testing procedures. These benefits include its low cost, the vast availability of X-ray facilities, non-invasiveness, less time consumption, and device affordability. Thus, X-ray imaging may be considered a better candidate for the mass, easy, and quick diagnosis procedure for a pandemic like COVID-19 considering the current global healthcare crisis.

The field of computational learning includes machine learning (ML) and deep learning (DL), aiming to detect meaningful patterns in data automatically and to solve problems, which are impossible (or impractical) to be represented by explicit algorithms. Traditional ML techniques have already been successfully applied to a diversity of pattern recognition and regression tasks. DL learns high-level abstractions in data by utilizing hierarchical architectures. It combines several layers of nodes to build up progressively more abstract representations of the data making it possible to learn concepts such as object categories directly from raw sensory data. The current success of DL is directly related to the spread of cheap, multi-processor graphics cards, or Graphics Processing Units (GPUs) that increase the speed and decrease the training time for creating a DL model. Considering the present pandemic situation, there is an appurtenant relationship between the detection of COVID-19 cases and chest X-ray image analysis and classification. In this work, an automatic diagnostic system has been developed using CNN which uses chest X-ray analysis results to diagnose whether a person is COVID-19- affected or normal. Preliminary analysis of this study has shown promising results in terms of its accuracy and other performance parameters to diagnose the disease in a cost-effective and time-efficient manner.

CNN is used in pattern recognition with superior feature learning capabilities, being a suitable model to deal with image data. Indeed, CNN is a dominant architecture of DL for image classification and can rival human accuracies in many tasks. CNN uses hierarchical layers of tiled convolutional filters to mimic the effects of human receptive fields on feed forward processing in the early visual cortex thereby exploiting the local spatial correlations present in images while developing robustness to natural transformations such as changes of viewpoint or scale.

A CNN-based model generally requires a large set of training samples to achieve good generalization capabilities. Its basic structure is represented as a sequence of Convolutional— Pooling—Fully Connected Layers possibly with other intermediary layers for normalization and/or dropout.

This model have used CNN with extra layers to improve the COVID-19 X-ray image classification accuracy. In neural networks, the CNN structure is specially designed to process the two-dimensional image tasks although it can also be used in one- and three-dimensional data. CNN is a type of DNN, inspired by the visual system of the human brain, and is most commonly used in the analysis of visual imagery. To train the CNN model, first, the dataset has been obtained from GitHub. Since the dataset obtained for training the model was very small in size and imbalanced, to solve the problem of having very-limited sized X-ray image dataset, it has been extended using data augmentation techniques to increase its size and also to make the model training feature rich. Image flipping and rotation at different angles have been used to generate more data. For dataset balancing in terms of proportion of images with different class labels, the dataset has been further extended with some more image instances of the minority class. After data augmentation and dataset balancing, the CNN model has been trained using a total of 800 images (400 COVID-19 and 400 normal) and then the model has been tested by using a test set.

The CNN model performance evaluation has then been done using different performance metrics. These metrics include accuracy, precision, sensitivity, specificity, ROC AUC, and F1 score.

The following are some of the key findings of this study:

* 1. CNN with extra convolutional layers (e.g., six layers have been used in the CNN proposed in this model) performs best in COVID-19 diagnosis.
  2. CNN models require a sufficient amount of images for efficient and more accurate image classification.
  3. Data augmentation techniques are very effective to improve the CNN model performance remarkably by generating more data from an existing limited size dataset.
  4. Data augmentation is also effective in image classification as it gives the ability of invariance to CNNs.
  5. The proposed CNN model performance has been proved statistically significant in the performance of other ML models.
  6. CNN-based diagnosis using X-ray imaging can be very effective for medical sector to handle the mass testing situations in pandemics like COVID-19.

# LITERATURE REVIEW

The analysis and detection of COVID-19 have been extensively investigated in the last few months. Firstly, we will addresses issues related to COVID-19 detection based on deep-learning approaches using CT scans and chest X-ray images. The second part reviews the related literatures to assess future estimates of the number of COVID-19 confirmations, recoveries, and deaths.

COVID-19 has now become a global pandemic owing to its rapid spread. It is very challenging to detect exposed persons because they do not show disease symptoms immediately. Thus, it is necessary to find a method of estimating the number of potentially infected persons on a regular basis to adopt the appropriate measures. AI can be used to examine a person for COVID-19 as an alternative to traditional time-consuming and expensive methods. Although there are several studies on COVID-19, this study focused on the use of AI in forecasting COVID-19 cases and diagnosing patients for COVID-19 infection through chest X-ray images. Several research areas have implemented AI (e.g. disease diagnoses in healthcare). One of the main advantages of AI is that it can be implemented in a trained model to classify unseen images. In this study, AI was implemented to detect whether a patient is positive for COVID-19 using their chest X-ray image. AI can also be used for forecasting (e.g., how the population will increase over the next 5 years) through existing evidence. Thus, predicting possibilities in the immediate future can help authorities to adopt the necessary measures.

The first concept involved studies related to the diagnosis of COVID-19, and the second involved studies related to the prediction of the number of people who will be infected in the coming days. The study analysis maintained that most of the existing models are poor and biased.

## 2.1) COVID-19 Diagnosis Using Deep Learning

The use of machine learning (ML) has been rapidly increasing in various fields including malware detection, mobile malware detection, medicine and information retrieval. In 2012, a modern ML system called deep learning was introduced, which is based on a Convolutional neural network (CNN). It won the ImageNet Large Scale Visual recognition challenge (ILSVRC), the world’s best-known computer-vision competition. Deep-learning algorithms enable computational models composed of multiple processing layers to learn data representation through several abstraction layers. They train a computer model to perform classification tasks directly from pictures, texts, or sounds. Deep-learning models feature high accuracies and can improve human output in certain instances.

## 2.2) X-Ray Diagnosis Using Deep Learning

X-ray machines use light or radio waves as radiation to examine the affected parts of the body because of cancers, lung diseases, bone dislocations, and injuries. Meanwhile, CT scans are used as sophisticated X-ray machines to examine the soft structures of active body parts for better views of the actual soft tissues and organs . The advantages of using X-rays over CT scans are that X-rays are quicker, safer, simpler, and less harmful than CT scans .Researchers proposed a CNN-based model to detect COVID-19 patients using 100 chest X-ray images, half of which belong to COVID-19 patients and the other half belong to healthy people. They evaluated three CNN models— ResNet-50, Inception-v3, and Inception-ResNet-v2—using five-fold cross- validation and reported that ResNet-50 had the best detection accuracy (98%). In a similar study conducted by Sethy and Behera, extracted features from chest X-ray images using a deep-learning algorithm and classified the images as either infected or healthy using a support vector machine (SVM).

The reasearchers employed 11 deep-learning models: AlexNet, VGG16, VGG19, GoogLeNet, ResNet-18, ResNet-50, ResNet-101, Inception-v3, Inception-ResNet-v2, DenseNet201, and XceptionNet. They collected two datasets—the first containing chest X-ray images of 25 infected patients and 25 non-infected patients and the other containing chest X-ray images of 133 infected patients (e.g. MERS, SARS, and ARDS patients) and 133 non-infected patients. They performed separate feature extractions on each dataset using various models and achieved a 95.38% accuracy with ResNet50 and SVM.

Furthermore, Hemdan et al. proposed a framework, called COVIDX-Net, that can assist radiologists in diagnosing COVID-19 patients using X-ray. They evaluated their framework using a dataset of 50 X-ray images divided into two classes: 25 COVID-19-positive images and 25 COVID19-negative images. The images used were resized to 224×224 pixels. The COVIDX-Net framework employs seven deep learning models: MobileNet, ResNet-v2, InceptionResNet-v2, Xception, Inception-v3, DenseNet, and modified VGG19. Their evaluation results indicate that the VGG19 and DenseNet models delivered comparable performances with an F-score of 91% for COVID-19 cases.

In addition, Hassanien et al. proposed a classification system that uses multi-level thresholding and an SVM to detect COVID-19 in lung X-ray images. Their system was tested on 40 contrast-enhanced lung X-ray images (15 healthy and 25 COVID-19-infected regions) with a resolution of 512×512 pixels. Their classification system achieved a sensitivity of 95.76%, a specificity of 99.7% and an accuracy of 97.48%.

* 1. **CT Scan Diagnosis Using Deep Learning**

The CT scan was developed by Godfrey Hounsfield and Allan Cormack in 1972. It utilizes an advanced X-ray technology to carefully diagnose delicate internal organs. CT scanning is quick, painless, non-invasive, and precise and can produce three-dimensional (3D) images. CT scans of internal organs, muscles, soft tissues, and blood vessels offer greater clarity than standard X- rays, especially for soft tissues and blood vessels. The main disadvantage of the CT scan is that it is expensive, compared to X-rays. The sensitivity and specificity of RT-PCR for COVID-19 detection have been criticized in several studies. Although RT-PCR is the standard method for this purpose, it generates a relatively large number of false negatives owing to several reasons, including methodological drawbacks, disease stages, and methods of obtaining the specimens, which delay disease diagnosis and control. Therefore, RT-PCR tests are not sufficient for assessing the disease status. Recent results have revealed that nucleic acid testing is not reliable and can only achieve an accuracy of 30–50%.

Jiang compared RT-PCR to CT scans and examined 51 patients (29 men and 22 women) with a history of travel to or residency in endemic areas and with severe respiratory and fever symptoms due to unknown causes. The researchers obtained a sensitivity of 98% in a non-contrast chest CT scan for the detection of COVID-19, compared to the initial RT-PCR sensitivity of 71%. Owing to the shortage of RT-PCR kits and the growing number of COVID-19 cases, it is important to introduce an automated detection system as an alternative diagnostic method to prevent the spread of COVID-19.

Meanwhile, Gozes employed a deep-learning approach to automatically identify COVID-19 patients and examine the disease burden quantification on CT scans using a dataset of CT scans from 157 foreign patients from China and the USA. Their proposed system analyses the CT scan at two distinct levels: subsystems A and B. Subsystem A performs a 3D analysis, and subsystem B performs a 2D analysis of each segment of the scan to identify and locate broader diffuse opacities, including ground-glass infiltrates (which have been clinically identified as representative of COVID-19).

To evaluate their system, the researchers applied Resnet-50-2 to subsystem B and obtained an area under the curve of 99.6%. The sensitivity and specificity were 98.2% and 92.2%, respectively. Moreover, Wang et al. developed a deep-learning approach for extracting information from CT scans. Their study included a collection of 453 CT scans from 99 patients.

They extracted 195 regions of interest (ROIs) of sizes ranging from 395×223 to 636×533 pixels from the CT scans of 44 COVID-19-positive pneumonia patients and 258 ROIs from those of 50 COVID-19-negative patients. They applied a modified network inception model and obtained an accuracy of 82.9% for the internal validation with a specificity of 80.5% and a sensitivity of 84%. The external testing dataset exhibited a total accuracy of 73.1% with a specificity of 67% and a sensitivity of 74%.

Fu et al. proposed a classification system based on ResNet-50 to detect COVID-19 and some other infectious lung diseases (bacterial pneumonia and pulmonary tuberculosis).

The researchers collected a dataset of 60,427 CT scans from 918 patients. 14,944 of these CT scans were from 150 COVID-19 patients and 15,133 from 154 non-COVID-19 viral pneumonia patients. They performed several tests for several lung diseases. The achieved accuracy, sensitivity, and specificity were 98.8%, 98.2%, and 98.9%, respectively.

Xu et al. reported that real-time RT-PCR has a low positive rate at the early stage of COVID-19. They developed an early screening model that uses deep-learning techniques for distinguishing COVID-19 pneumonia from influenza (a viral pneumonia) and stable cases using pulmonary CT images. A dataset of 618 CT samples was obtained for the analysis, and the images were classified as COVID-19, influenza, and other cases using ResNet-18 and ResNet-based methods. The researchers employed a noisy or Bayesian function to differentiate the infected images and obtained a detection accuracy of 86.7%.

* 1. **COVID-19 Infection Prediction Using Machine Learning Techniques**

ML is the science of training machines using mathematical models to learn and analyze data. Once ML is implemented in a system, the data are analyzed, and interesting patterns are detected. The validation data are then categorised according to the patterns learned during the learning process. As COVID-19 infection has rapidly spread worldwide and international action is required, it is important to develop a strategy to estimate the number of potentially infected people on a regular basis to adopt the appropriate measures.

Currently, decision-makers rely on certain decision-making statistics such as imposing lockdowns on infected cities or countries. Therefore, ML can be used to predict the behaviors of new cases to stop the disease from spreading. Li et al. developed a prediction model using ML algorithms to combat COVID-19 in mainland China and in other infected countries in the world. The researchers developed a model to estimate the number of reported cases and deaths in mainland China and in the world. The data used to build the models were collected between 20 January 2020 and 1 March 2020.

# TECHNOLOGY AND METHODS

Our proposed deep learning- based Covid-19 detection comprises of several phases. The phases are summarized in the following five steps:

**Step 1:** Collect the chest X-Ray images for the dataset from Covid-19

patients and healthy persons.

**Step 2:**  Split the dataset into two sets- a training set and a validation set.

**Step 3:**  Data Preparation and Filtering

**Step 4:** Represent the images in a feature space and apply Deep Learning.

**Step 5:** Evaluate the performance of the model on the validation dataset.

# Dataset

Chest imaging is commonly used in medicine, and it plays an important role in the detection of COVID-19. Through the diagnosis of chest imaging, medical staff can more accurately grasp the imaging modal characteristics of COVID-19 cases, such as multiple small patchy shadows and interstitial changes in the early stage, which are obvious outside the lungs. It then develops into multiple ground glass and infiltration shadows in both lungs. In severe cases, lung consolidation and pleural effusion are rare. It has important guiding value for accurately judging the condition and its development, formulating treatment plans, and evaluating prognoses. There are many COVID-19 datasets, but the number of samples is small.

Two types of datasets were used in the evaluation, the original dataset (without augmentation) and the augmented dataset. The dataset contained the following:

1. a healthy dataset containing chest X-ray images of healthy persons and,
2. a Covid-19 dataset containing chest X-ray images of Coivd-19 patients.

The experiment collected 4099 COVID-19 chest X-rays, consists of 3278 in a training set and 821 in a validation (test) set.

|  |  |  |  |
| --- | --- | --- | --- |
|  | | **Training Set** | **Validation Set** |
| **Covid** | **578** | **462** | **116** |
| **Normal** | **550** | **440** | **110** |
| **Total** | **1128** | **902** | **226** |



Figure 1.1: X-Ray images of Covid Positive



Figure 1.2: X-Ray images of Covid Negative

1. **Data Preparation and Filtering**

In this methodology, we try to filter images from both the datasets and only use ..filtered images further for training and validating the model. In our dataset, there are images of Covid19, Pneumonia, and SARS so we filter out these images and took only the Covid-19 images, also there were many different views of the images present but we selected posterior interior view of the X-ray images.

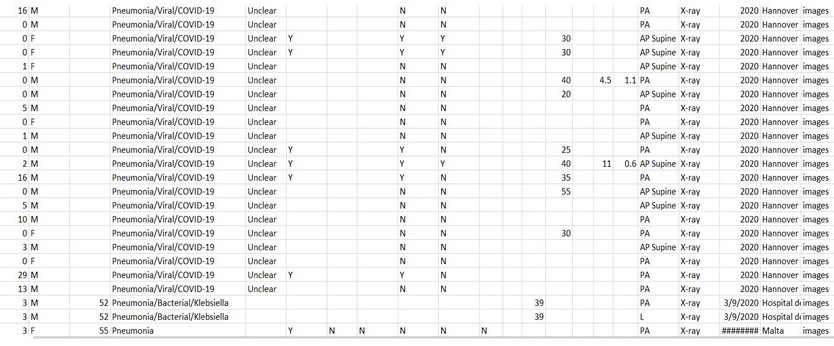


Figure 2: A sample Covid-19 Dataset from Github

# Training model

## 3.3.a) Convolutional Neural Network (CNNs):

The CNNs are inspired by visual system of human brain. The idea behind the CNNs thus is to make the computers capable of viewing the world as humans view it. This way CNNs can be used in the fields of image recognition and analysis, image classification, and natural language processing. CNN is a type of deep neural networks which contain the convolutional, max pooling, and non-linear activation layers.

CNNs are inspired by biological processes. They’re based on some [cool research done by Hubel](http://klab.tch.harvard.edu/academia/classes/Neuro230/2014/readings/reading_assignment2_gk1852.pdf) [and Wiesel in the 60s](http://klab.tch.harvard.edu/academia/classes/Neuro230/2014/readings/reading_assignment2_gk1852.pdf) regarding vision in cats and monkeys. The pattern of connectivity in a CNN comes from their research regarding the organization of the visual cortex. In a mammal’s eye, individual neurons respond to visual stimuli only in the receptive field, which is a restricted region. The receptive fields of different regions partially overlap so that the entire field of vision is covered. This is the way that a CNN works!

The convolutional layer, considered as a main layer of a CNN, performs the operation called **“convolution”** that gives CNN its name. Kernels in the convolutional layer are applied to the layer inputs. All the outputs of the convolutional layers are convolved as a feature map. The Rectified Linear Unit (ReLU) is used as the activation function with a convolutional layer which is helpful to increase the non-linearity in input image, as the images are fundamentally non-linear in nature. Thus, CNN with ReLU in the current scenario is easier and faster. Since the ReLU is zero for all negative inputs, it can be defined as:

## z = max(0, i)

Here, the function implies that the output z is zero for all negative value and positive value remains the constant.

The pooling layer or sub sampling layer is also an important building block of CNN. On each feature map extracted through the convolution layer, the pooling layer operates independently. To minimize overfitting and the number of extracted features, it decreases the spatial size of the feature map and returns the important features.

Pooling can be the max, average, and sum in the CNN model. We have used max pooling because others may not identify the sharp features easily as compared to max pooling.

The dropout layer with a 25% dropout rate has also been used, which drops the neurons during the training chosen at random to reduce the overfitting problem. Towards the last stage of the CNN, there is a flatting layer to convert the output of convolutional layers into a single- dimensional feature vector. In other words, the flattening layer arranges all the pixel data output produced by convolutional layers in one vector. After flattening, the vector data is given as an input to the next layers of the CNN called fully connected layers or dense layers. In a fully connected layer, each neuron of the previous layer is directly connected to each of the neurons in its next layer. The main functionality of dense layers is to take flattened output results from the convolution and pooling layers and as input and classify the image to a specific class label.

Each value of the flattened feature set represents the probability of a feature belonging to a specific class. Thus, on the basis of these probabilities, the fully connected network with dense layers finally drives the classification decision. We have build our Convolutional Neural Network Model with the help of TensorFlow Keras library.

# 

Figure 3: Layers in CNN

# MODEL ARCHITECTURE:-

The proposed CNN model consists of 14 layers in which 4 are convolutional (Conv2D), 3 max pooling layers, 4 dropout layers, 2 activation function layers, and 1 flatten layer. CNN model input image shape is (224, 224, 3), i.e., 224-by-224 RGB image. In all Con2D layers, a 3 × 3 size kernel has been used but the filter size after every two Con2D layers increases. At the 1st layer of Con2D, 32 filters have been used to learn from input, and the 2nd and 3rd layers of Con2D use 64 filters, and at the 4th, 128 filters have been used. After each Con2D layer, the max pooling layer with 2 × 2 pooling size has been used, 1st activation layer has been used with the ReLU function, and 2nd activation layer has been used with the sigmoid function.

The dropout layer has been used with 25% dropout rate. The output of 128 output neurons of the final Con2D layer is followed by max pooling, activation, and dropout layer. Since the final pooling and convolutional layer gives a three-dimensional matrix as output, to flatten the matrix, a flattening layer has been used which converts them into a vector that will be input for dense layers.

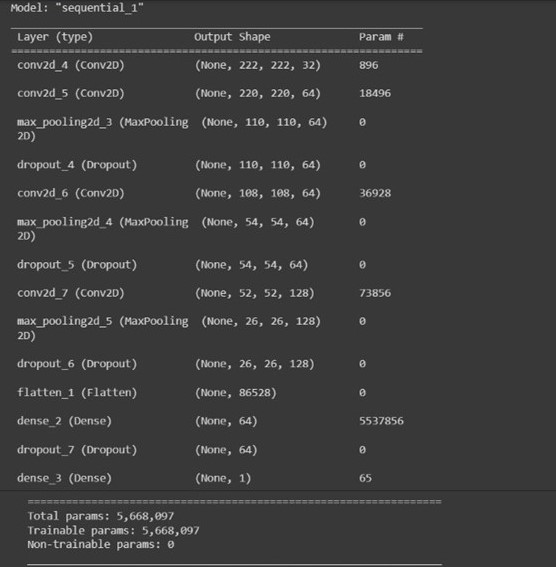


Figure 4: Table showing Total params, Total Trainable and Non-trainable params

1. **LIBRARIES USED:-**

**3.5.a) TensorFlow**

TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications.

# 3.5.b) Keras

Keras is a high-level, deep learning API developed by Google for implementing neural networks. It is written in Python and is used to make the implementation of neural networks easy. It also supports multiple backend neural network computation.

Keras is relatively easy to learn and work with because it provides a python frontend with a high level of abstraction while having the option of multiple back-ends for computation purposes. This makes Keras slower than other deep learning frameworks, but extremely beginner- friendly. Keras allows you to switch between different back ends.

# 3.5.c) NumPy

NumPy is a Python package. It stands for 'Numerical Python'. It is a library consisting of multidimensional array objects and a collection of routines for processing of array.

Numeric, the ancestor of NumPy, was developed by Jim Hugunin. Another package Numarray was also developed, having some additional functionalities. In 2005, Travis Oliphant created NumPy package by incorporating the features of Numarray into Numeric package. There are many contributors to this open source project.

Using NumPy, a developer can perform the following operations −

* Mathematical and logical operations on arrays.
* Fourier transforms and routines for shape manipulation.
* Operations related to linear algebra. NumPy has in-built functions

for linear algebra and random number generation.

# 3.5.d) OS Module

The OS module in Python provides functions for interacting with the operating system. OS comes under Python’s standard utility modules. This module provides a portable way of using operating system-dependent functionality. The \*os\* and \*os.path\* modules include many functions to interact with the file system.The OS comes under Python's standard utility modules. This module offers a portable way of using operating system dependent functionality.

# 3.5.e) Open-CV

OpenCV is a huge open-source library for computer vision, machine learning, and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. It can process images and videos to identify objects, faces, or even the handwriting of a human. When it is integrated with various libraries, such as NumPy which is a highly optimized library for numerical operations, then the number of weapons increases in your Arsenal i.e., whatever operations one can do in NumPy can be combined with OpenCV.

# 3.5.f) Pandas

Pandas is a Python library for data analysis. Started by Wes McKinney in 2008 out of a need for a powerful and flexible quantitative analysis tool, pandas has grown into one of the most popular Python libraries. It has an extremely active community of contributors.

Pandas is built on top of two core Python libraries—matplotlib for data visualization and NumPy for mathematical operations. Pandas acts as a wrapper over these libraries, allowing you to access many of matplotlib's and NumPy's methods with less code.

For instance, pandas' .plot() combines multiple matplotlib methods into a single method, enabling you to plot a chart in a few lines. Before pandas, most analysts used Python for data munging and preparation, and then switched to a more domain specific language like R for the rest of their workflow. Pandas introduced two new types of objects for storing data that make analytical tasks easier and eliminate the need to switch tools: Series, which have a list-like structure, and DataFrames, which have a tabular structure.

# VISUALIZATION:-

**3.6.a) Matplotlib**

Matplotlib is an amazing visualization library in Python for 2D plots of arrays. Matplotlib is a multi-platform data visualization library built on NumPy arrays and designed to work with the broader SciPy stack. It was introduced by John Hunter in the year 2002. One of the greatest benefits of visualization is that it allows us visual access to huge amounts of data in easily digestible visuals. Matplotlib consists of several plots like line, bar, scatter, histogram etc.

# 3.6.b) Seaborn

Seaborn is an open-source Python library built on top of [matplotlib](https://www.section.io/engineering-education/matplotlib-visualization-python/). It is used for data visualization and exploratory data analysis. Seaborn works easily with dataframes and the Pandas library. The graphs created can also be customized easily.

It is closely integrated with pandas data structures in Python. Visualization is the central part of Seaborn which helps in exploration and understanding of data.

# 3.6.c) HTML

HTML (HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation ([CSS](https://developer.mozilla.org/en-US/docs/Web/CSS)) or functionality/behavior ([JavaScript](https://developer.mozilla.org/en-US/docs/Web/JavaScript)).

"Hypertext" refers to links that connect web pages to one another, either within a single website or between websites. Links are a fundamental aspect of the Web. By uploading content to the Internet and linking it to pages created by other people, you become an active participant in the World Wide Web.

HTML uses "markup" to annotate text, images, and other content for display in a Web browser.

# 3.6.d) CSS

Cascading Style Sheets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.

CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.

CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

# 3.6.e) jQuery

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to- use API that works across a multitude of browsers. With a combination of versatility and extensibility, jQuery has changed the way that millions of people write JavaScript. The jQuery made it easy to select DOM elements, negotiate them and modifying their content by using cross-browser open source selector engine called **Sizzle**.

# 3.6.f) Flask Web Framework

Flask is considered more [Pythonic](http://blog.startifact.com/posts/older/what-is-pythonic.html) than the [Django](https://www.fullstackpython.com/django.html) web framework because in common situations the equivalent Flask web application is more explicit. Flask is also easy to get started with as a beginner because there is little boilerplate code for getting a simple app up and running. Flask is a micro [web framework](https://en.wikipedia.org/wiki/Web_framework) written in [Python](https://en.wikipedia.org/wiki/Python_(programming_language)). It is classified as a [micro-framework](https://en.wikipedia.org/wiki/Microframework) because it does not require particular tools or libraries.It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

Applications that use the Flask framework include [Pinterest](https://en.wikipedia.org/wiki/Pinterest) and [LinkedIn.](https://en.wikipedia.org/wiki/LinkedIn) Flask was created by [Armin Ronacher](https://en.wikipedia.org/wiki/Armin_Ronacher) of Pocoo, an international group of Python enthusiasts formed in 2004. According to Ronacher, the idea was originally an [April Fool's](https://en.wikipedia.org/wiki/April_Fool%27s) joke that was popular enough to make into a serious application.

The **route()** function of the Flask class is a decorator, which tells the application which URL should call the associated function**. app.route(rule, options)**

* The **rule** parameter represents URL binding with the function.
* The **options** is a list of parameters to be forwarded to the underlying Rule object.

Finally the **run()** method of Flask class runs the application on the local development server. app.run(host, port, debug, options)

We have used Flask to represent the web framework of our model. It basically shows the X-ray image that we are selecting and then the corresponding result of COVID19 Negative or COVID19 Positive.

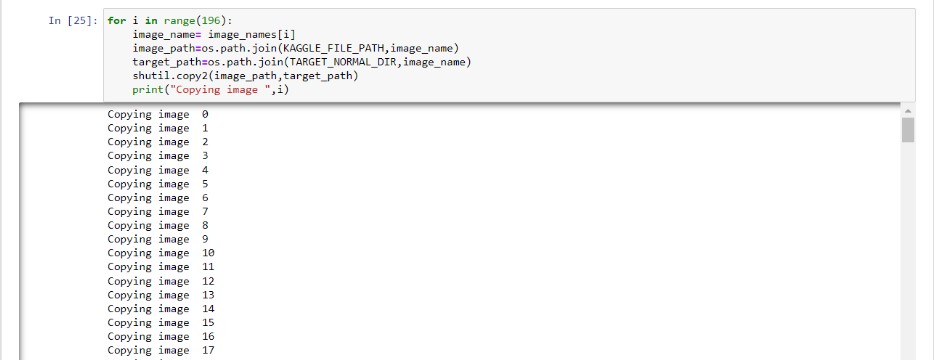
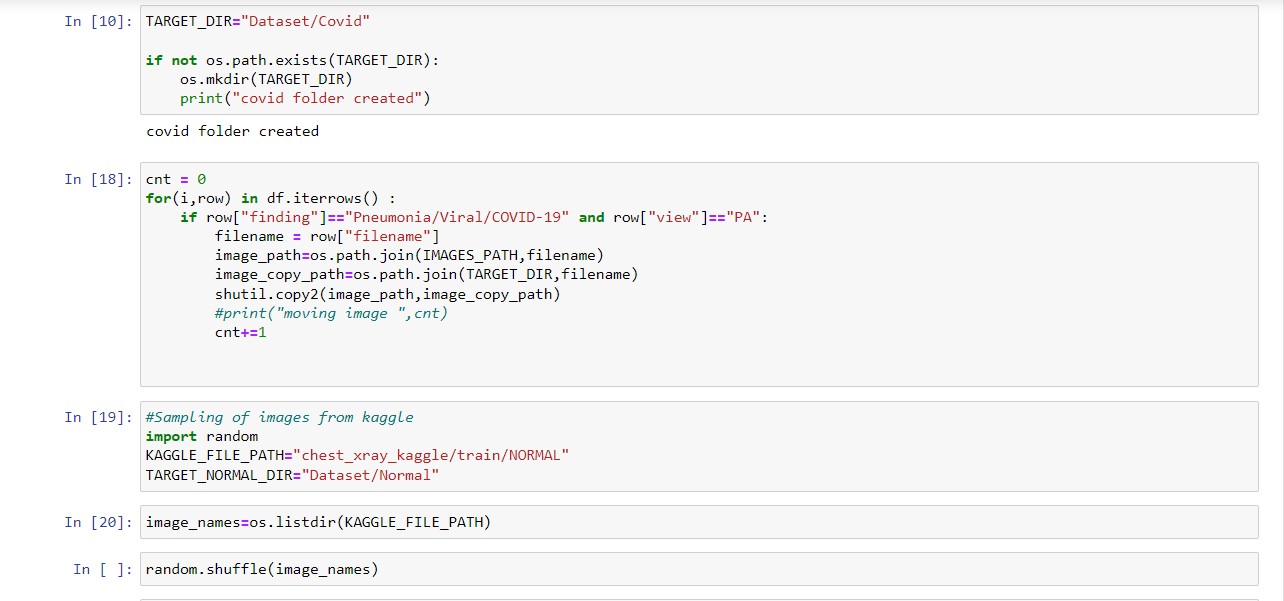
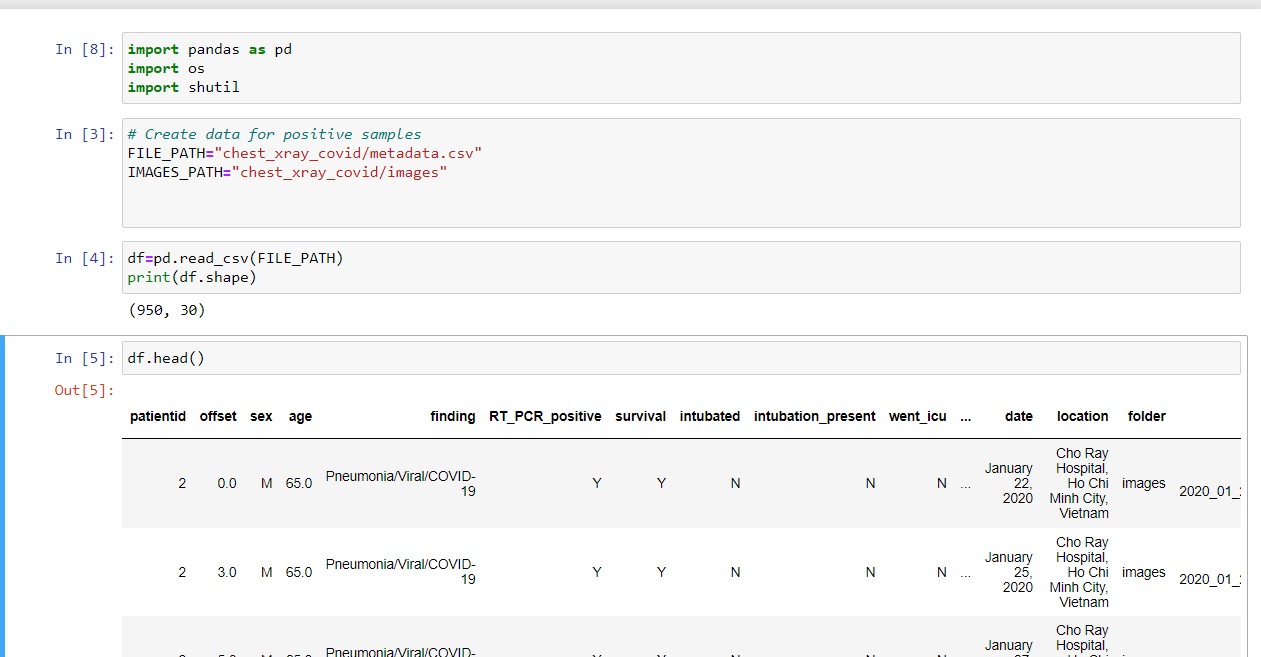
# 3.6.g) Pillow

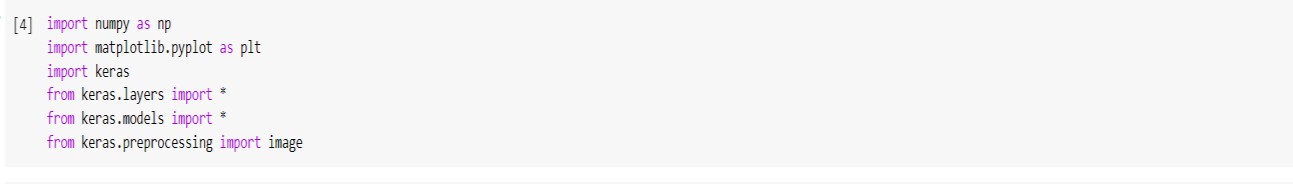
Pillow is a Python Imaging Library (PIL), which adds support for opening, manipulating, and saving images. The current version identifies and reads a large number of formats. Write support is intentionally restricted to the most commonly used interchange and presentation formats.

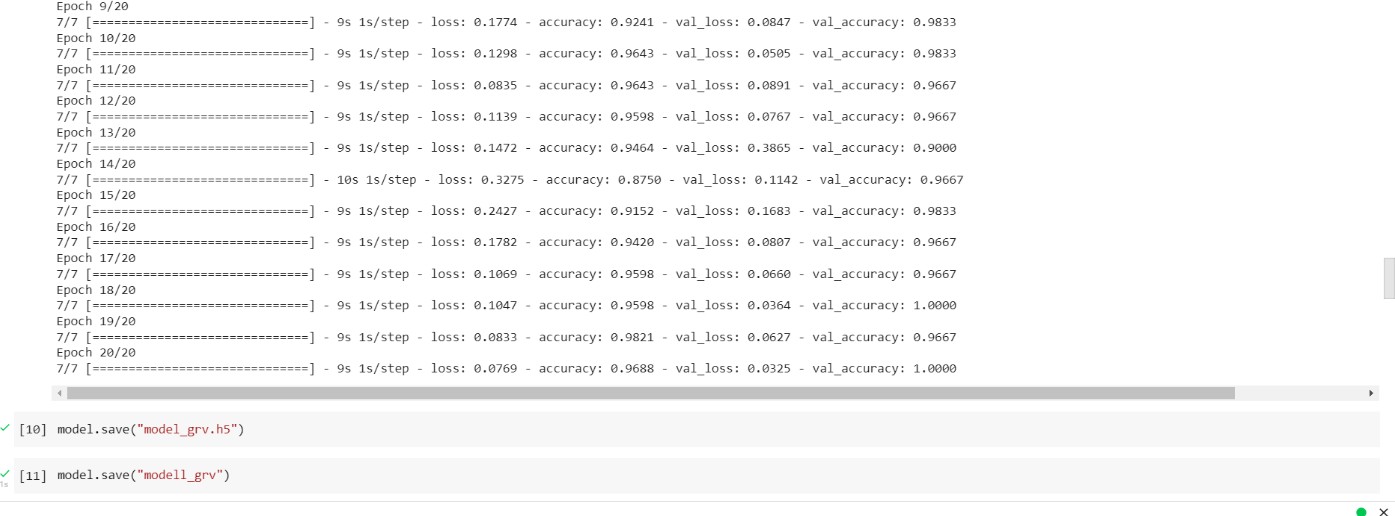
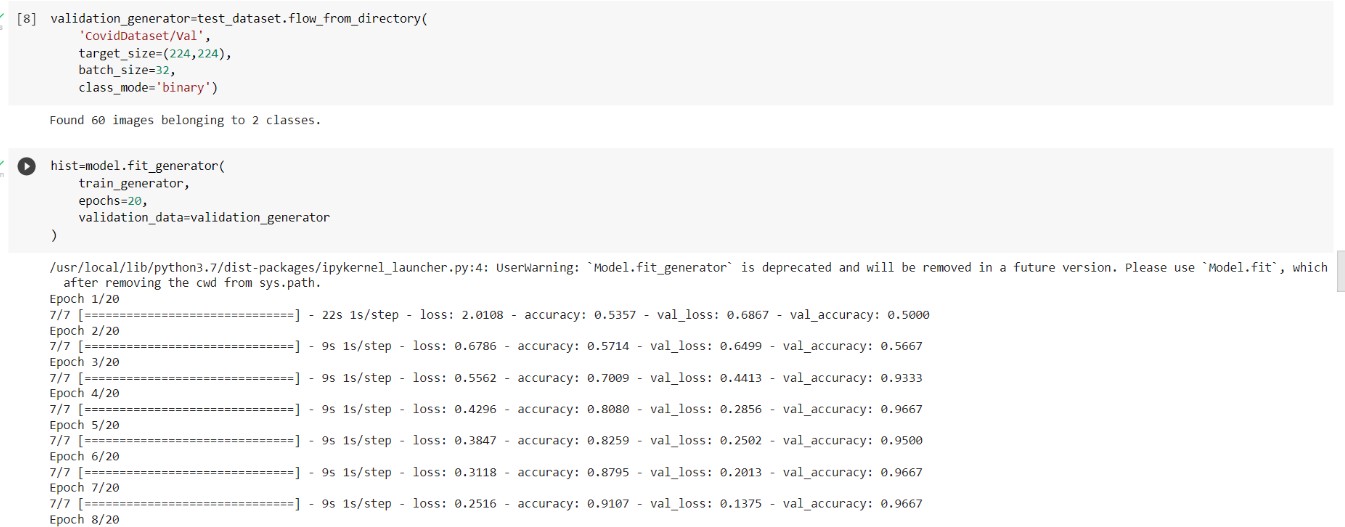
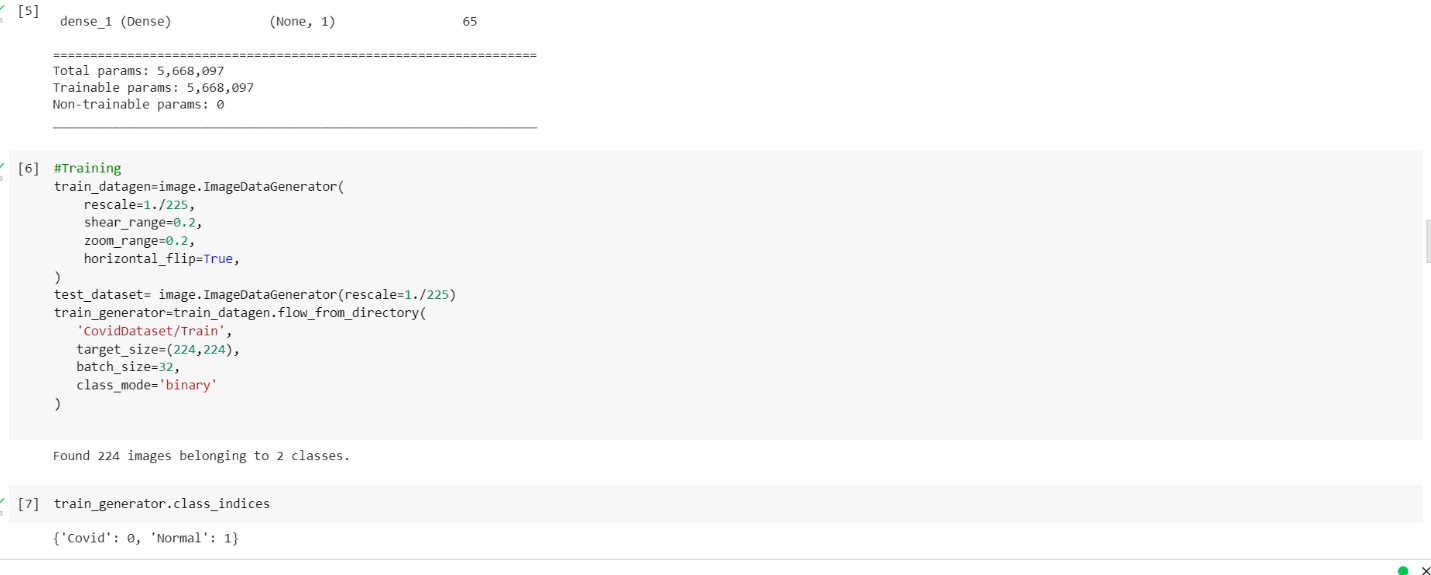
This is the library which is a fork of the Python Imaging Library (PIL) to handle the opening and manipulation of images in many formats in Python. The Python Imaging Library adds image processing capabilities to your Python interpreter. This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities.

In today’s digital world, we come across lots of digital images. In case, we are working with Python programming language, it provides lot of image processing libraries to add image processing capabilities to digital images.

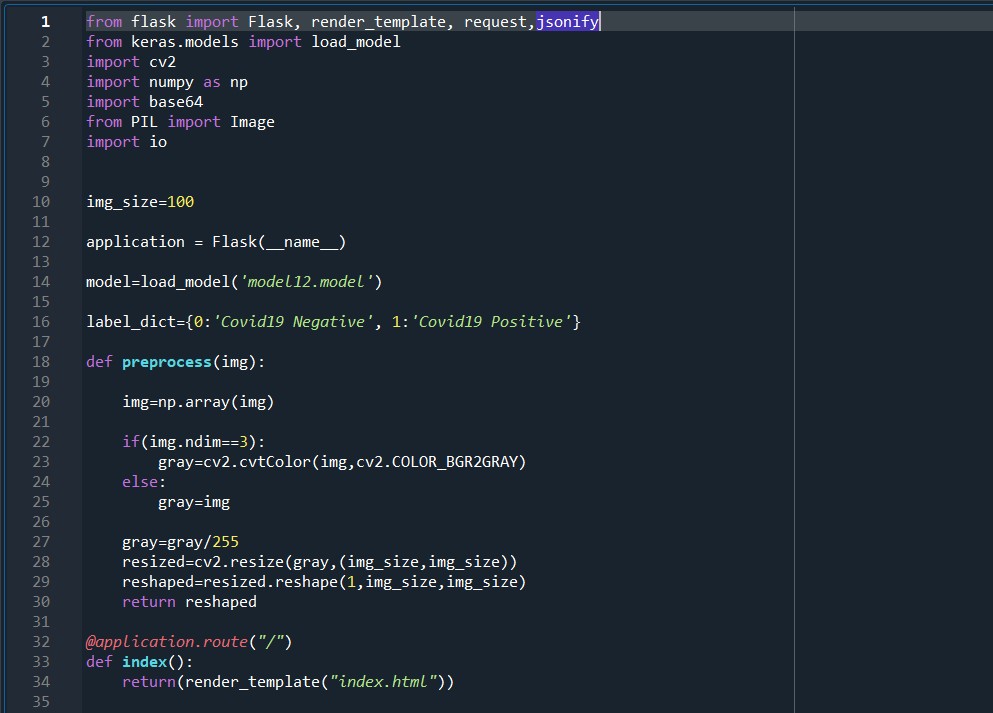
# DATA PREPARATION



1. **MODEL CREATION**

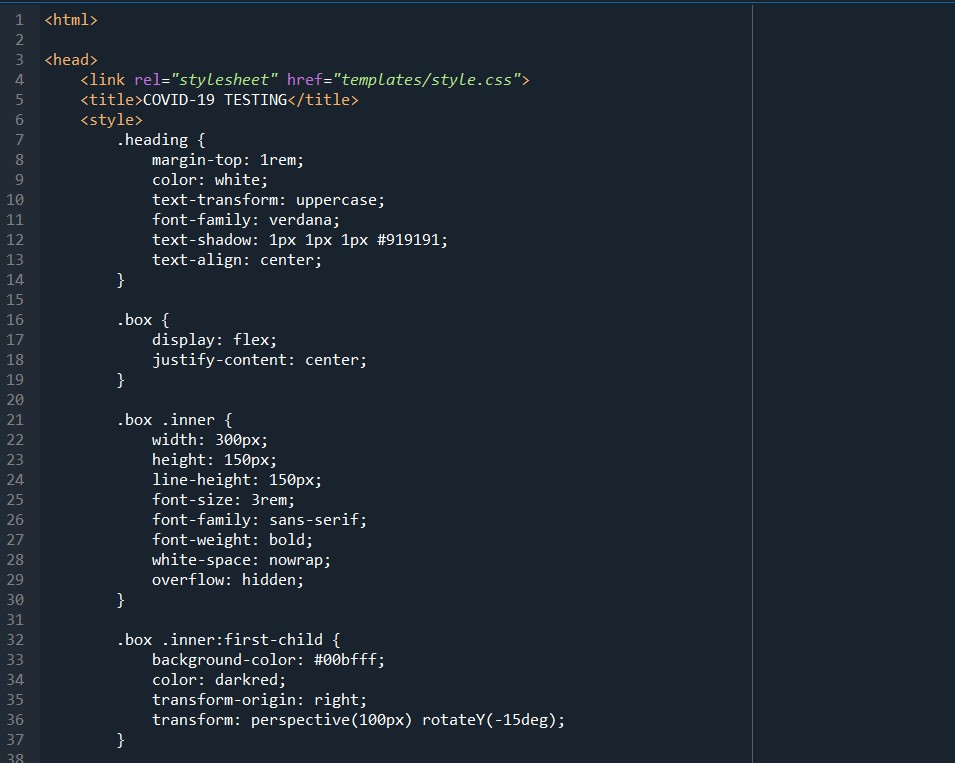


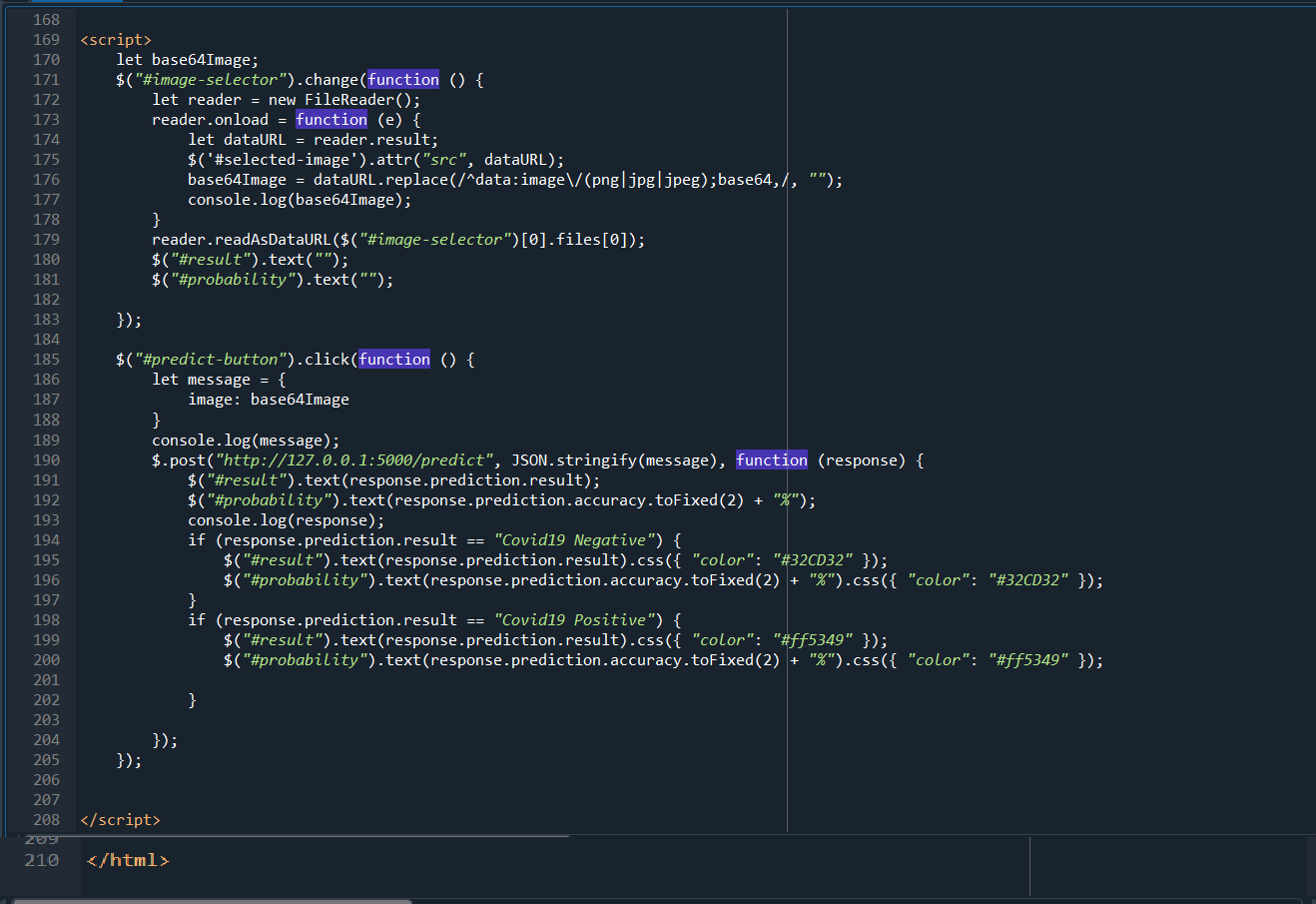
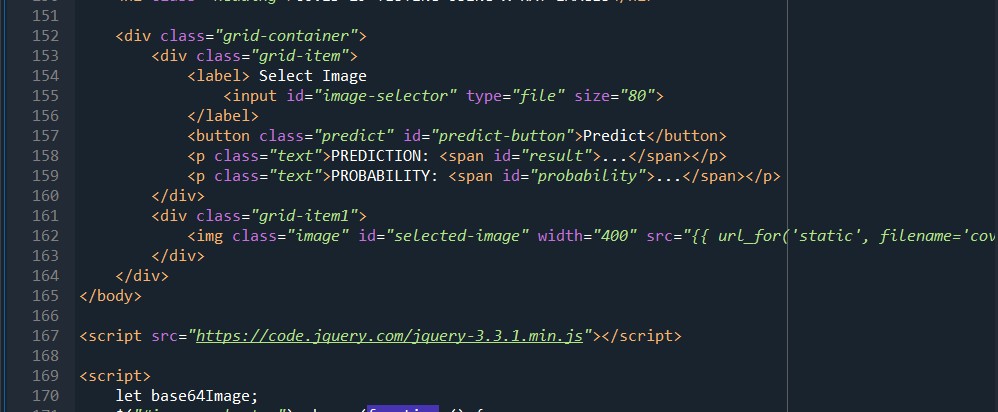
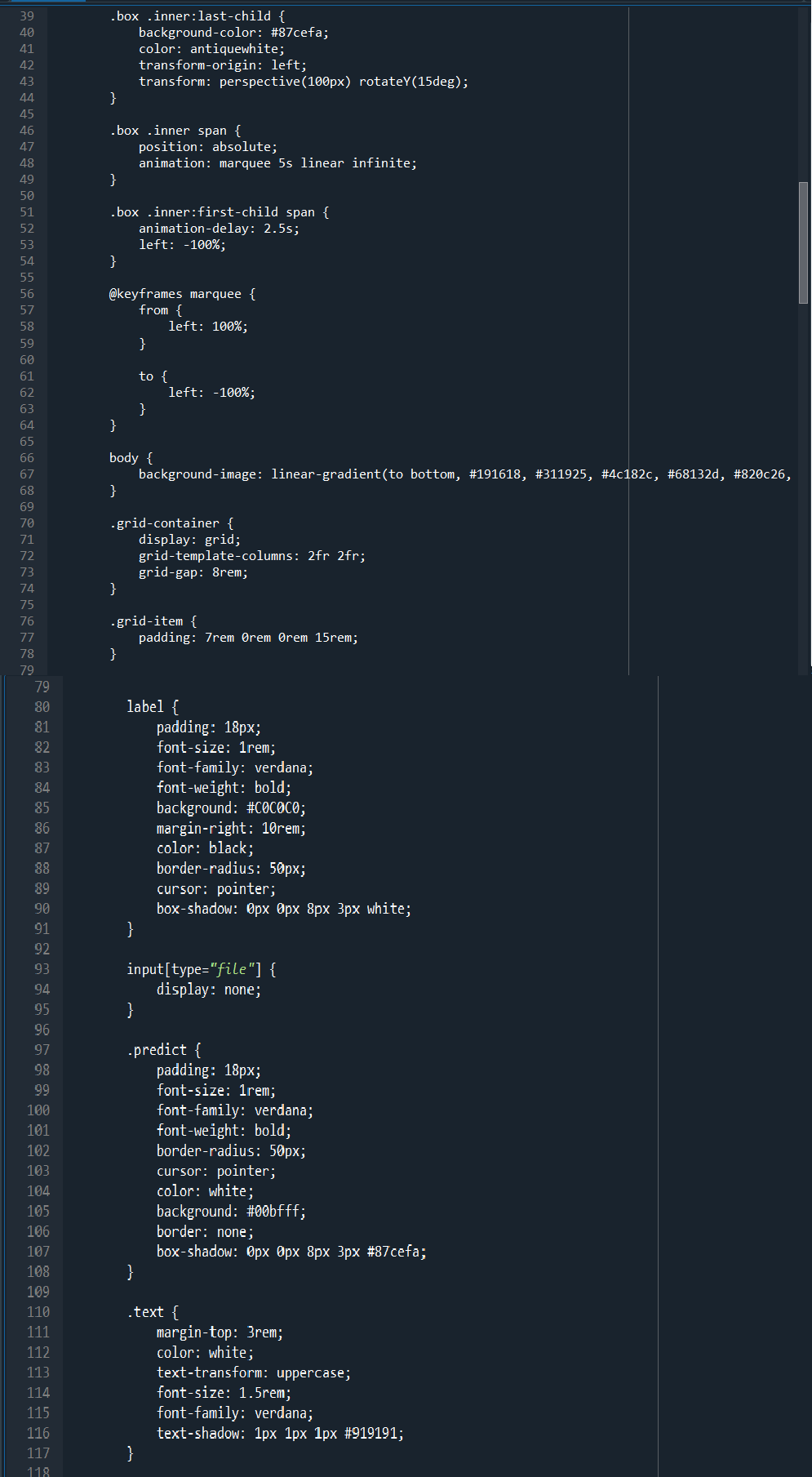
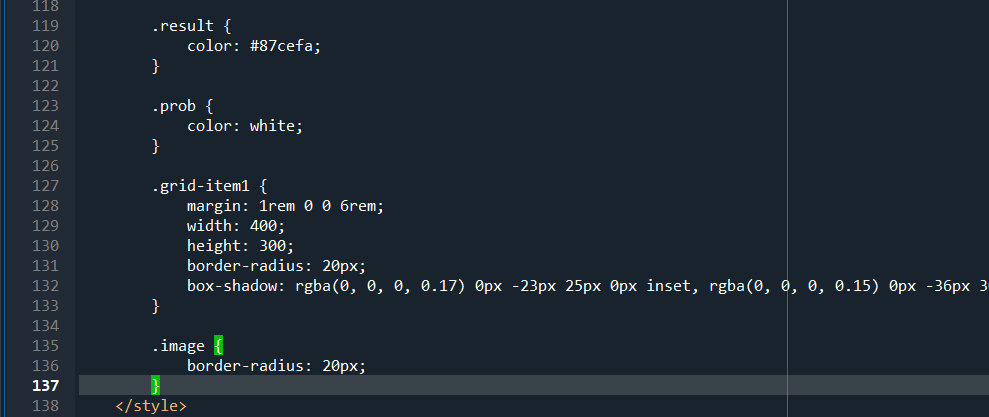
# WEB:-

 **3.9.a) application.py**



# 3.9.b) Index.html (HTML+ CSS + jQuery)





1. **RESULTS AND DISCUSSION**

In this section, the Accuracy metrics of Convolutional Neural Networks (CNNs) are discussed. As in the above section we successfully trained and tested the model.

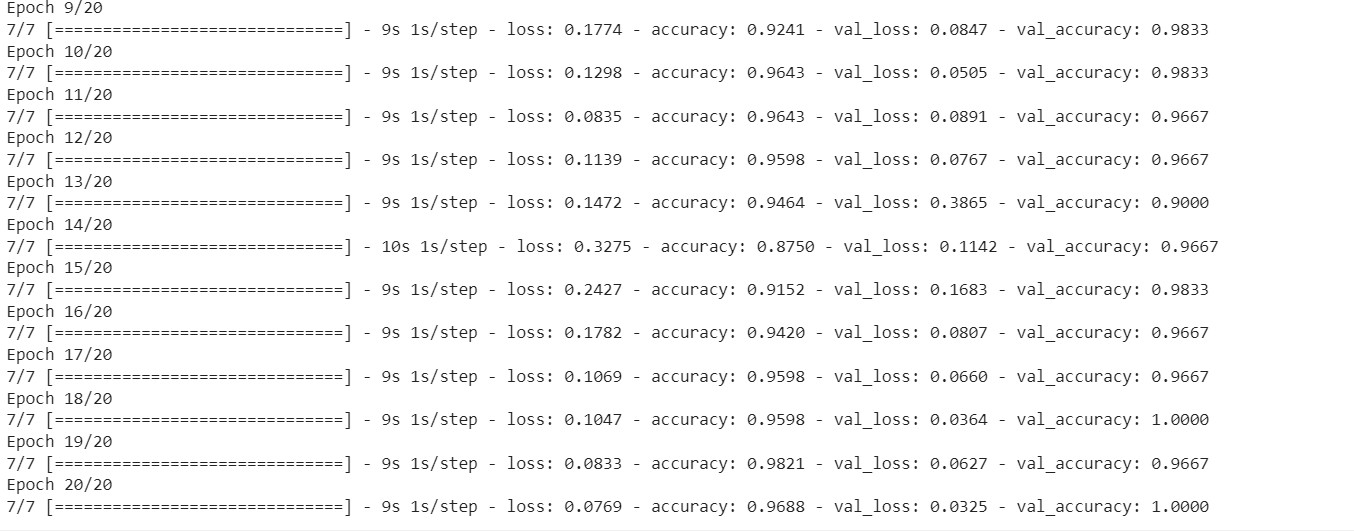


Figure 5: Test loss, test accuracy, train loss and train accuracy obtained after testing

After validation/test we got the validation accuracy of 98% from the starting to ending of training the validation accuracy was improved. The loss percentage decreased gradually.

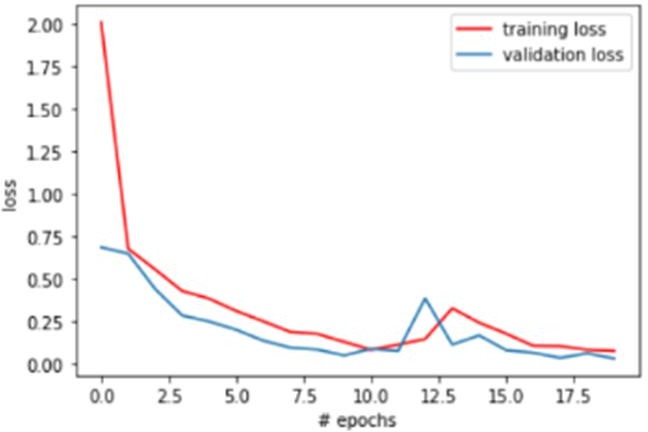


Figure 6: Graph of training loss Vs. Validation loss

Above graph indicates a gradual decrease in loss per step in validation. The blue line indicates the training Loss and the Red line indicates validation loss.



Figure 7: Training Accuracy Vs. Validation Accuracy

Above graph shows the Accuracy of our model which shows the best results. The blue lines indicate the training accuracy and the orange line indicates validation accuracy, where the best validation accuracy being 98% and the best training accuracy was 96%. This happens after you use Dropout, since the behavior once training and testing is completely different. We have also applied the confusion matrix to our model. The confusion matrix is one of the best accuracy matrices. The confusion matrix is an easier way to measure the performance of the classification problem.

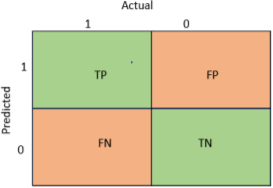
A confusion matrix is a two-dimensional table with actual and predicted values containing “True Positives (TP)”, “True Negatives (TN)”, “False Positives (FP)”, “False Negatives (FN)” like this:-

Figure 8: Format of a confusion matrix

The TP (True Positives) and TN (True Negatives) need to be a higher number because those only indicates correct classification. The remaining FP and FN need to be very less number because Those are false predictions.

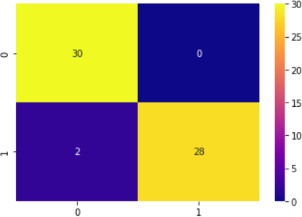


Figure 9: A confusion matrix showing result of our model

We have given 30 COVID images and 30 Normal images for the validation dataset. The model correctly predicted 30 images as COVID positive and 0 image as False positive. Out of 30 normal images, the model predicted 28 images as normal and 2 as False Negative.

The model got trained and tested, which results in 95% training accuracy which shows the best results. We have also tested the model by giving individual images, and it shows top-notch results.

With the help of python modules i.e. PIL and a flask web framework, a python script is created by which one image is taken as input in web framework through local systems and then we get the desired result as COVID POSITIVE or COVID NEGATIVE.

The expected results are shown below:

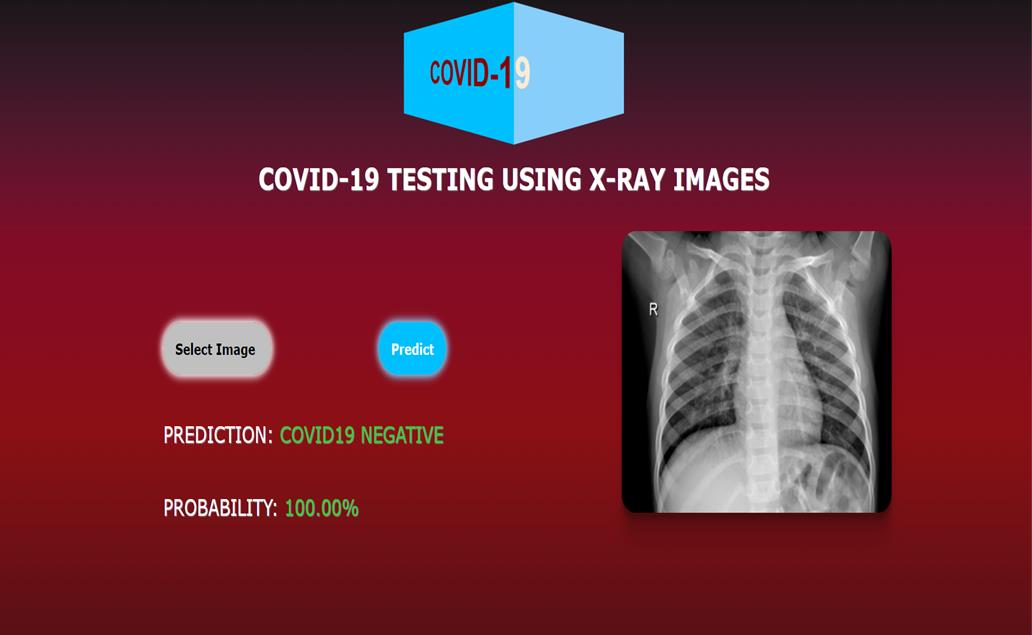


Figure 10.1: Output as Covid Negative

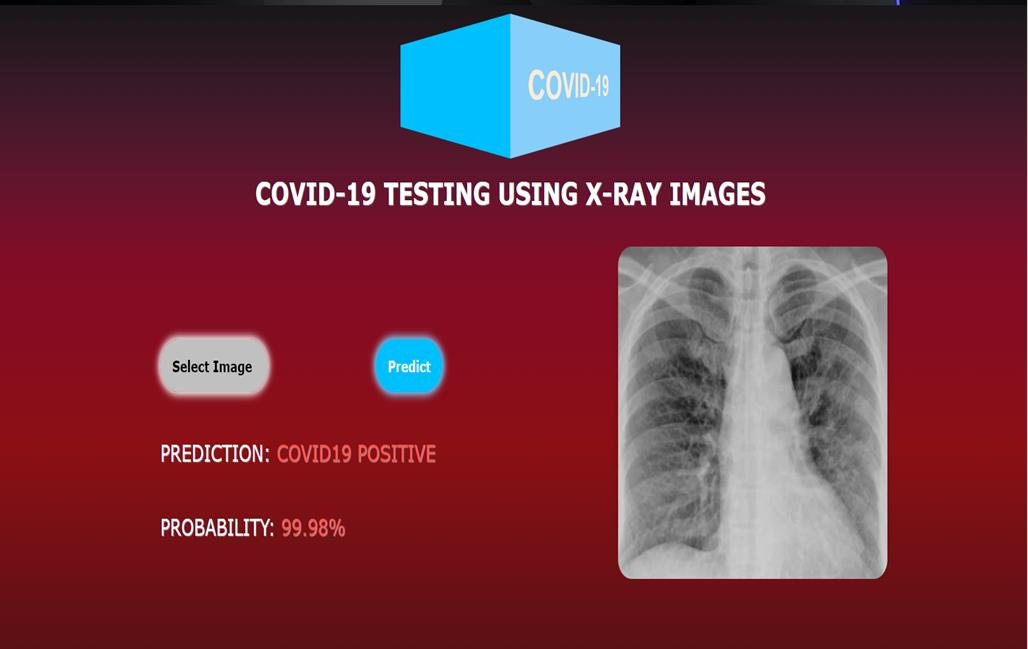


Figure 10.2: Output as Covid Positive

# CONCLUSION

The rapid spread of COVID-19 across the world and the increasing number of deaths require urgent actions from all sectors. Future prediction of potential infections will enable authorities to tackle the consequences effectively. Furthermore, it is necessary to keep up with the number of infected people by performing regular check-ups, and it is often vital to quarantine infected people and adopt medical measures. Additionally, attention should be given to several other factors to curb the spread of COVID-19, such as the environmental effects and the similarities among the most affected areas, and careful measures should be adopted.

This model describes an alternative data-driven diagnostic method for COVID-19 to support clinical decision-making processes from images and based on CNN models. In this model we have detected COVID-19 using two databases from Kaggle and Github with X-rays images giving 99% accuracy.

The proposed method has major limitations. The experiment only applies to X-ray images, and not CT images, because X-ray images are RGB and CT images are grayscale. This experiment can only be used to classify COVID-19 patients and healthy people.

The classification accuracy can also be improved using more data samples, more convolutional layers, and using more complex models. It also enables us to cater to much wider community which will result in more data samples.

# REFERENCES

1. Aijaz Ahmad Reshi , Furqan Rustam , Arif Mehmood , Abdulaziz Alhossan, Ziyad Alrabiah, Ajaz Ahmad , Hessa Alsuwailem, and Gyu Sang Choi “An Efficient CNN Model for COVID-19 Disease Detection Based on X-Ray Image Classification”, <https://www.hindawi.com/journals/complexity/2021/6621607/>
2. Areej A.Wahab Ahmed Musleh, Ashraf Yunis Maghari “COVID-19 Detection in X-ray Images using CNN Algorithm”,https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9319286
3. Shrinjal Singh, Piyush Sapra, Aman Garg, Dinesh Kumar Vishwakarma “CNN based Covid-aid: Covid 19 Detection using Chest X-ray” ,[https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9418407](https://ieeexplore.ieee.org/stamp/stamp.jsp?tp&arnumber=9418407)
4. Pillalamarry Mahesh, Yakkala Gnana Prathyusha , Botlagunta Sahithi , S Nagendram “Covid-19 Detection from Chest X-Ray using Convolution Neural Networks”,<https://iopscience.iop.org/article/10.1088/1742-6596/1804/1/012197/pdf>
5. Deploy Machine Learning Model using Flask ,<https://www.geeksforgeeks.org/deploy-machine-learning-model-using-flask/>
6. Flask webserver - adding HTML and CSS ,<https://pythoninformer.com/python-libraries/flask/adding-html-and-css/>
7. B. Kayalibay, G. Jensen, and P. van der Smagt, “CNN-based segmentation of medical imaging data,” 2017, <http://arxiv.org/abs/1701.03056>
8. Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen, “Medical image classification with convolutional neural network,” in Proceedings of the 2014 13th International Conference on Control Automation Robotics & Vision (ICARCV), pp. 844–848, Singapore, December 2014
9. D. J. Cennimo, “Coronavirus disease 2019 (COVID-19) clinical presentation,” vol. 8, pp. 101489–101499, 2020, [https:// emedicine.medscape.com/article/2500114-clinical#b2, 2020. Online](%20https:/%20emedicine.medscape.com/article/2500114-clinical%23b2,%202020.%20Online)
10. V. K. Ayyadevara, “Gradient boosting machine,” in Pro Machine Learning AlgorithmsSpringer, Berlin, Germany, 2018.
11. R. Ramteke and K. Y. Monali, “Automatic medical image classification and abnormality detection using k-nearest neighbour,” International J
12. P. K. Sethy, S. K. Behera, P. K. Ratha, and P. Biswas, “Detection of coronavirus disease (COVID-19) based on deep features and support vector machine,” 2020.
13. I. D. Apostolopoulos and T. A. Mpesiana, “Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks,” Physical and Engineering Sciences in Medicine, vol. 43, pp. 635–640, 2020.
14. N. N. Das, N. Kumar, M. Kaur, V. Kumar, and D. Singh, “Automated deep transfer learning-based approach for detection of COVID-19 infection in chest X-rays,” IRBM, 2020.