International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

DETECTION OF COVID-19 FROM RADIOGRAPHS

V Kruthika[1], V Sai Prasanna[2], T Jhansi[3], T Monika Singh[4]
Department of Computer Science and Engineering, Stanley College of Engineering and Technology for Women, Telangana, India

Abstract. COVID-19 pandemic has caused millions of deaths in more than 150 countries around the world. Detecting this disease from radiography and radiology images is perhaps one of the fastest ways to diagnose the patients. This application can be used by anyone to detect whether a person is COVID-19 affected or not according to his/her chest x-ray. The process of detecting whether a person is COVID-19 affected or not can be automated by training a deep learning model so that it can predict the given x-ray accurately. This is a Supervised learning problem and classification algorithms are used. The dataset used is collected from GitHub (Covid-19 positive x-rays) and Kaggle (Covid-19 negative x-rays). CNNs are extensively used and combined with consolidated machine learning methods, such as k-Nearest Neighbor, Bayes, Random Forest, multilayer perceptron (MLP), and support vector machine (SVM). The results will show if the person is COVID affected by examining his chest x-ray. For working on this project, we have used python in Jupiter notebook.

Keywords: Succumbed, Consolidated, COVID-19, Chest X-Ray Image, image identification, Detecting.

1 Introduction

1.1 About

The root of the coronavirus word is Greek (κορώνη), i.e., crown or halo, which refers to the virus appearance, means Viral infection, under an electron microscope which is similar to a royal crown. That's why coronavirus is also referred to as the crowned virus. The COVID-19 emerged as an epidemic disease in China, Wuhan City, in December 2019. Today, this has altered to a pandemic as a dangerous public health problem all around the world. The COVID-19 also has other names, e.g., SARS-COV-2 virus. This virus is a type of large-family viruses divided into four types including α-coronavirus, β-coronavirus, δ-coronavirus, and γ-coronavirus.

Coronavirus disease (COVID-19) is an infectious disease caused by a newly discovered coronavirus. Most people infected with the COVID-19 virus will experience mild to moderate respiratory illness and recover without requiring special treatment. Older people, and those with underlying medical problems like cardiovascular disease, diabetes, chronic respiratory disease, and cancer are more likely to develop serious illness. In the current situation outside an efficient tool is required which predicts the presence of COVID-19 virus and give more awareness about it. COVID-19 detector is a web application which predicts COVID-19 virus from the chest x-ray of a person.

1.2 Objective

One of the critical factors behind the rapid spread of COVID-19 pandemic is a lengthy clinical testing time. The imaging tool, such as Chest X-ray (CXR), can speed up the identification process. Therefore, our objective is to develop an automated CAD system for the detection of COVID-19 samples from healthy and covid cases using Chest X-Ray images, and to create an image classification model that can detect Chest X-Ray scans that belong to one of the two classes with a reasonably high accuracy. Our objective in this project is to create an image classification model that can predict Chest X-Ray scans that belong to one of the three classes with a reasonably high accuracy.

1.3 Scope

The COVID-19 pandemic and resulting 'lockdown'/social distancing measures has resulted in a considerable change in practice in how we work with individuals who have sustained an acquired brain injury (and their families/support networks). We are determining the support network as a significant other that is close to the client, but may not be a family member (e.g., friend, support worker). To obtain the experiences of clients who have sustained an acquired brain injury, their families and case managers following the recent COVID-19 pandemic.

The application can be used by anyone to detect whether a person is COVID-19 affected or not according to his/her chest x-ray. The process of detecting from x-rays is a Supervised learning problem and classification algorithms are used. The dataset used is collected from GitHub (Covid-19 positive x-rays) and Kaggle (Covid- 19 negative x-rays). CNNs are extensively used and combined with consolidated machine learning methods, such as k-Nearest Neighbor, Bayes, Random Forest, multilayer perceptron (MLP), and support vector machine (SVM). Random Forest gives greater accuracy when compared with the other algorithms. The results will show if the person is COVID affected by examining his chest x-ray.

1.4 Advantages

- It is Flexible.
- It is very important for both doctors and patients to decrease the diagnostic time.
- Reduces diagnostic cost.
- The application can be used by one who knows basic system operations and basic English

1.5 Disadvantages

- The results predicted are not 100% accurate and may vary when new training data is added. So, the results obtained should be properly examined before proceeding further. There is no authentication.
- We get result accurately only when the posteroanterior view of X-Ray is uploaded.
- The user should not totally rely on the prediction made by this application. The results predicted are only based on the previously trained results. Hence predicted results should not be treated as final.

- Here there is no user authentication everybody is free to use the application. However the report generated should not be disclosed and distributed outside.
- To get accurate results the user should upload a valid x-ray. A valid x-ray for this application is a posteroanterior view of an x-ray which is also called pa view of an x-ray. Giving a right x-ray will improve the accuracy of the result predicted.

1.6 Hardware and Software Requirements

1.6.1. Software Requirements

- Operating System: Windows XP/7/10.
- Domain: Deep Learning.
- Coding Language: Python.
- Development Kit: Flask Framework.
- Libraries like pandas, NumPy, Keras, and TensorFlow etc. are used.
- The web application is deployed using the Flask framework

1.6.2. Hardware Requirements

- System: Intel Core i3 2.20 GHZ.
- Hard Disk: 500GB.
- High Speed Internet.
- RAM: Minimum 2 GB.

1.7. Functional Requirements

X-Ray type: The user should upload a valid x-ray with a posteroanterior (PA) view, else incorrect predictions can be made.
Predicted Results: The results predicted should not be taken 100% accurate as they are predicted according to the previous x-rays trained.

1.7.1 General Awareness

1. DESCRIPTION AND PRIORITY

Description: This feature of General awareness helps the user to know more about the symptoms seen, precaution information, vaccine information etc.
Priority: Medium Priority

2. STIMULUS/RESPONSE SEQUENCES

• The user can go to any of the 3 sections Symptoms, Precautions, and covid test

• If the users go to the symptoms section, he/she can view all the information about the symptoms seen when a person is COVID-19 affected.

• If the user goes to the precautions section, he/she can view all the information about the precautions to be taken to stay safe.

• If the user goes to the vaccine info section, he/she can view all the information about the COVID-19 vaccine.

3. Functional Requirements

The results shown in this feature are purely static and the information shown is purely based on the information available on the internet and is taken from trusted sources.

1.8. Other Non- Functional Requirements

1.8.1. Performance Requirements

To get accurate results the user should upload a valid x-ray. A valid x-ray for this application is a poster anterior view of an x-ray which is also called PA view of an x-ray. Giving a right x-ray will improve the accuracy of the result predicted.

1.8.2. Safety Requirements

The user should not totally rely on the prediction made by this application. The results predicted are only based on the previously trained results. Hence predicted results should not be treated as final.

1.8.3. Security Requirements

There is no user authentication required; everybody is free to use the application However, the report generated should not be disclosed and distributed outside.

1.8.4. Software Quality Attributes

The results predicted are not 100% accurate and may vary when new training data is added. So, the results obtained should be properly examined before proceeding further.

2 Literature Survey

2.1 Existing System

The following are types of existing coronavirus tests:

Antigen test: In this case a special swab is used to take sample from nose or throat where the patient typically has to go to hospital to have this test performed. COVID-19 antigen test detects coronavirus proteins in the mouth and throat. This test determines whether a patient is currently infected with COVID-19. Antigen testing is a suitable ‘first line of defense’ test against COVID-19, particularly in individuals who have COVID symptoms,

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

because it is a relatively simple procedure, and is quicker than other approaches like PCR testing; It is less accurate than PCR testing,

Nasal Aspirate: In this case a saline solution will be injected into nose and then the sample is taken with the light suction and the main dis advantage is it is not widely available.

RT- PCR: Real time reverse transcriptase polymerase chain reaction RT-PCR is time taking because the sample needed to go to lab for testing. RT PCR tests have been the most sought-after tests to check for COVID-19 infections since the beginning of the pandemic in India. Many specialists consider it to be the most effective test to detect COVID-19 virus in human cells.

2.2 Proposed System

In this project we are using supervised and classification algorithms for detecting whether a person is affected with Covid-19 or not. Firstly, we will train a machine with two types of datasets. One dataset contains positive X-Ray images and another dataset contains negative X-Ray images. In this project we have used Deep Learning algorithms like CNN for classification.

3 Proposed Architecture

3.1 System Architecture

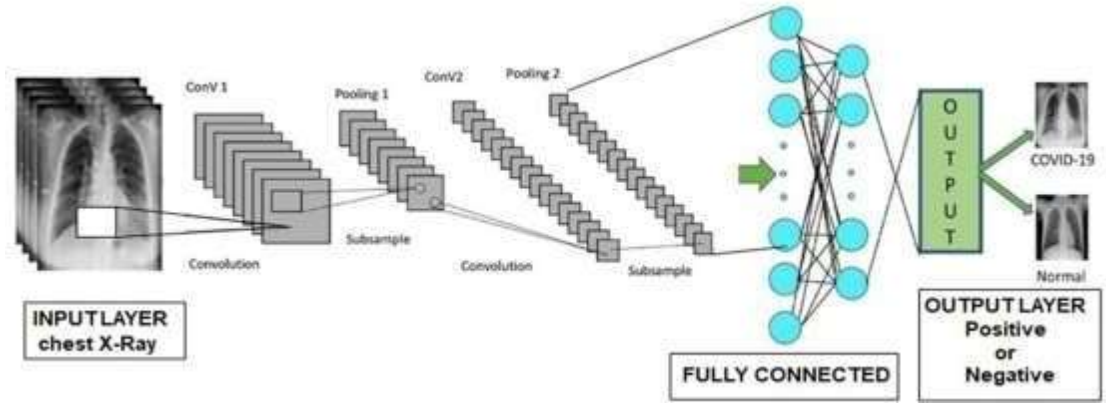


Fig 3.1: Architecture

CONV 1: This layer creates a convolution kernel that is convolved with the layer input over a single spatial (or temporal) dimension to produce a tensor of outputs. If use_bias is true, a bias vector is created and added to the outputs. Finally, if activation is not none, it is applied to the outputs as well.

When using this layer as the first layer in a model, provide an input shape argument (tuple of integers or None, e.g. (10, 128) for sequences of 10 vectors of 128-dimensional vectors, or (None, 128) for variable-length sequences of 128-dimensional vectors.

CONV2D: 2D convolution layer (e.g., spatial convolution over images).

This layer creates a convolution kernel that is convolved with the layer input to produce a tensor of outputs. If use_bias is true, a bias vector is created and added to the outputs. Finally, if activation is not none, it is applied to the outputs as well.

When using this layer as the first layer in a model, provide the keyword argument input_shape (tuple of integers or None, does not include the sample axis), e.g., input_shape= (128, 128, 3) for 128x128 RGB pictures in data_format="channels_last". You can use none when a dimension has variable size.

Pooling Layer: The pooling operation involves sliding a two-dimensional filter over each channel of feature map and summarizing the features lying within the region covered by the filter. For a feature map having dimensions nh x nw x nc, the dimensions of output obtained after a pooling layer is

(nh - f + 1) / s x (nw - f + 1)/s x nc

nh - height of feature map -> nw - width of feature map -> nc - number of channels in the feature map -> f - size of filter -> s - stride length

A common CNN model architecture is to have a number of convolution and pooling layers stacked one after the other.

Types of Pooling Layers

Max Pooling

Max pooling is a pooling operation that selects the maximum element from the region of the feature map covered by the filter. Thus, the output after max-pooling layer would be a feature map containing the most prominent features of the previous feature map.

Average Pooling

Average pooling computes the average of the elements present in the region of feature map covered by the filter. Thus, while max pooling gives the most prominent feature in a particular patch of the feature map, average pooling gives the average of features present in a patch.

Global Pooling

Global pooling reduces each channel in the feature map to a single value. Thus, a nh x nw x nc feature map is reduced to 1 x 1 x nc feature map. This is equivalent to using a filter of dimensions nh x nw i.e., the dimensions of the feature map.

Further, it can be either global max pooling or global average pooling

.

Fully-Connected: A Convolutional Neural Network (CNN) is a type of neural network that specializes in image recognition and computer vision tasks.

CNNs have two main parts:

– A convolution/pooling mechanism that breaks up the image into features and analyzes them

– A fully connected layer that takes the output of convolution/pooling and predicts the best label to describe the image

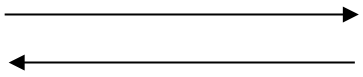
Fully Connected Layer is simply, feed forward neural networks. Fully Connected Layers form the last few layers in the network.The input to the fully connected layer is the output from the final Pooling or Convolutional Layer, which is flattened and then fed into the fully connected layer.

3.2 UML Concepts

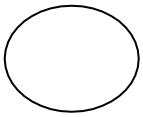
Data Flow Diagrams

Data Flow Diagram can also be termed as bubble chart. It is a pictorial or graphical form, which can be applied to represent the input data to a system and multiple functions carried out on the data and the generated output by the system. A graphical tool accustomed describe and analyze the instant of knowledge through a system manual or automatic together with the method, stores of knowledge, and delays within the system. The transformation of knowledge from input to output, through processes, is also delineate logically and severally of the physical elements related to the system. The DFD is also known as a data flow graph or a bubble chart. The Basic Notation used to create a DFD's are as follows:

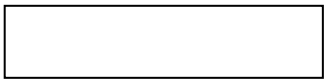
➢ Dataflow:



➢ Process:



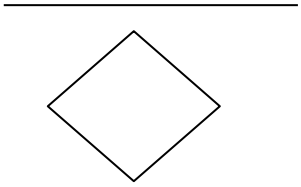
➢ Source:



➢ Data Store:

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

➢ Rhombus: decision



3.2.1 Level-1

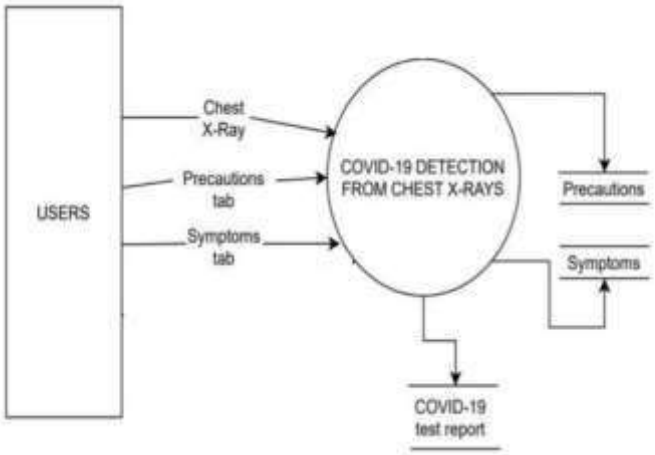


Fig 3.2.1: Level-1 Data Flow Diagram

This is the level-1 Data flow diagram which shows everything about the work flow of the application.

3.2.2 Level-2

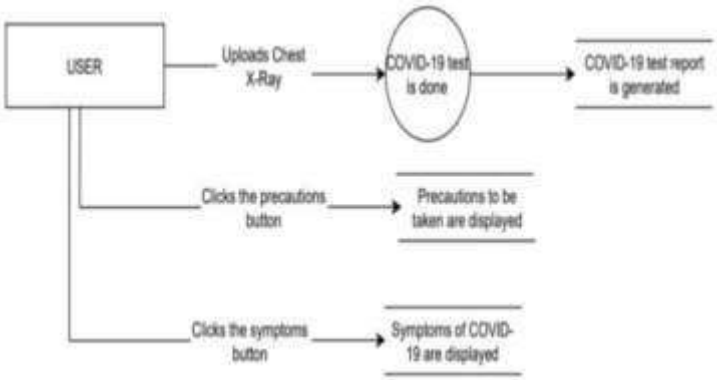


Fig 3.2.2: Level-2 Data Flow Diagram

This is the Level-2 Data flow diagram which shows the in-detail explanation about the users and the web interface.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

3.2.3 ER DIAGRAM

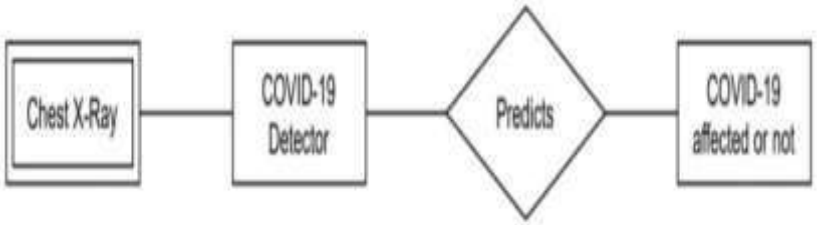


Fig 3.2.3: ER Diagram

This ER diagram shows about the various entities and their relationships.

UML DIAGRAMS

The Unified Modeling Language allows the software engineer to express an analysis model using the modeling notation that is governed by a set of syntactic semantic and pragmatic rules. A UML system is represented using five different views that describe the system from distinctly different perspective. Each view is defined by a set of diagrams, which is as follows. User Model View This view represents the system from the user's perspective. The analysis representation describes a usage scenario from the end-user's perspective. Structural Model view in this model the data and functionality are arrived from inside the system. This model view models the static structures. Behavioral Model View It represents the dynamic of behavioral as parts of the system, depicting the interactions of collection between various structural elements described in the user model and structural model view. Implementation Model View In this the structural and behavioral as parts of the system are represented as they are to be built.

3.2.4 USE CASE DIAGRAM

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams as well.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

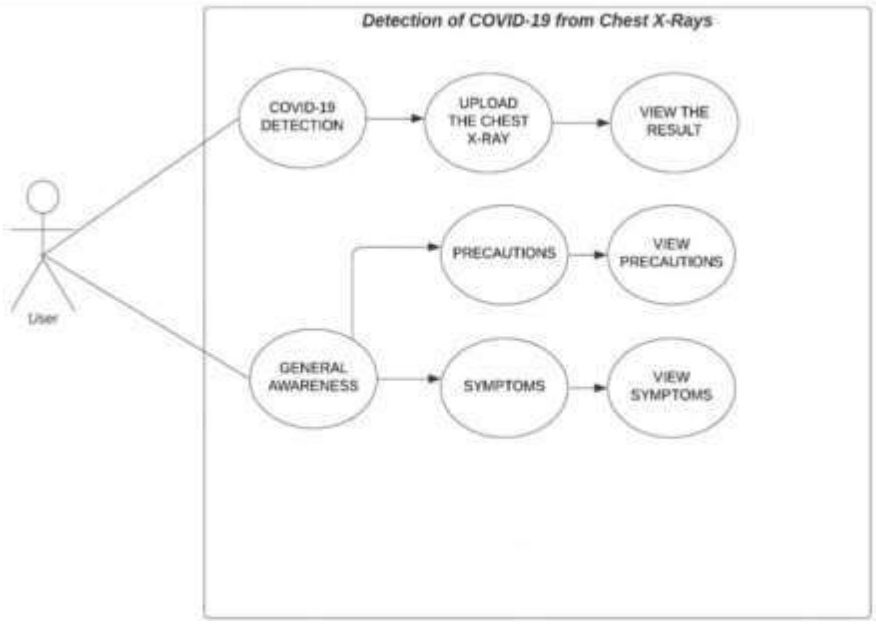


Fig 3.2.4: Use Case Diagram

Use case diagram shows the detailed explanation of all the scenarios in the application. Here the user can go for COVID-19 test by uploading his/her chest x-ray or go for general awareness to know more about the symptoms and precautions to be taken.

3.2.5 CLASS DIAGRAM

The class diagram is the main building block of object-oriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed. A class with three sections, in the diagram, classes is represented with boxes which contain three parts: The upper part holds the name of the class The middle part contains the attributes of the class.

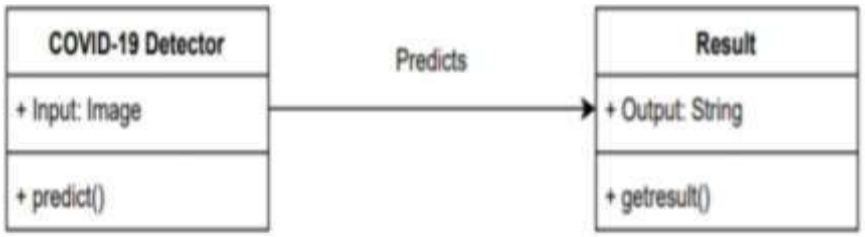


Fig 3.2.5: Class Diagram

The class diagram is used to show different classes of the application along with their attributes. Here we have Detector class and Result class where detector class needs an attribute image to predict the result class with

attribute report which is a string.

3.2.6 SEQUENCE DIAGRAM

A sequence diagram is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development.

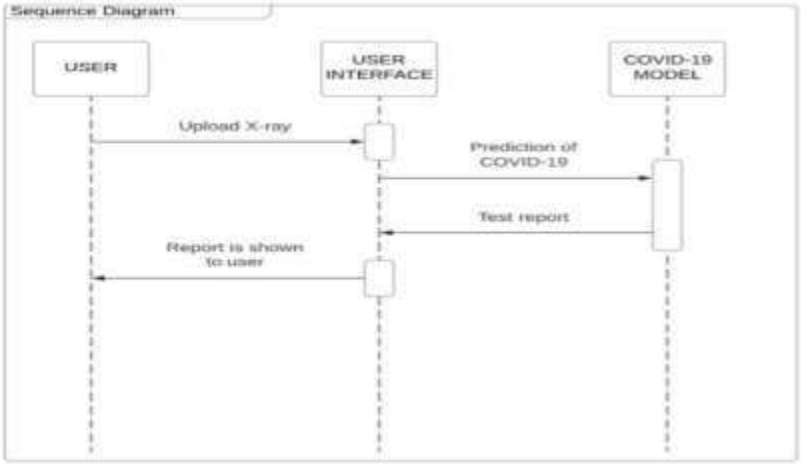


Fig 3.2.6: Sequence Diagram

Sequence diagrams are the interaction diagrams that detail how operations are carried out. As shown above the user uploads an x ray which is then sent to the covid-19 model which in turn predicts the presence of COVID-19 and sends the report to the user.

3.2.7 ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

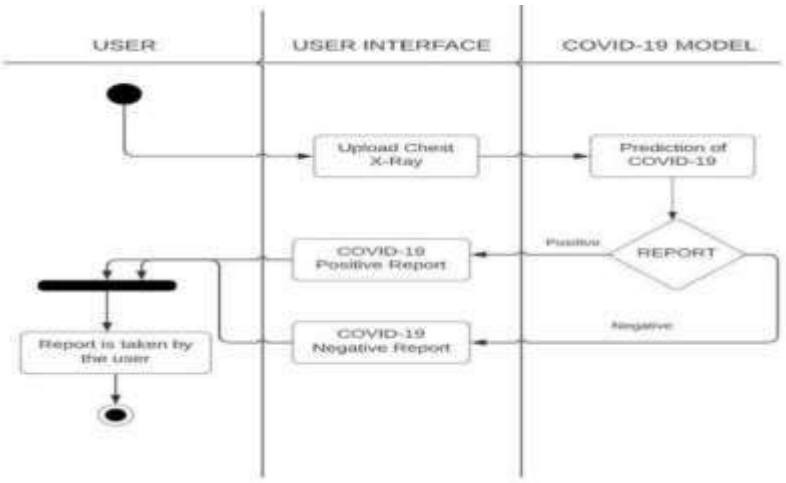


Fig 3.2.7: Activity Diagram

Activity diagram is presenting a series of actions, flow of control in a system in an application. As shown above the user initially uploads the chest x-ray and the covid-19 model predicts the presence of virus and if the report is positive the user interface generates a positive report, and if the report is negative a covid-19 negative report is generated. Then the report is viewed by the user.

3.2.8 Component Diagram

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities.

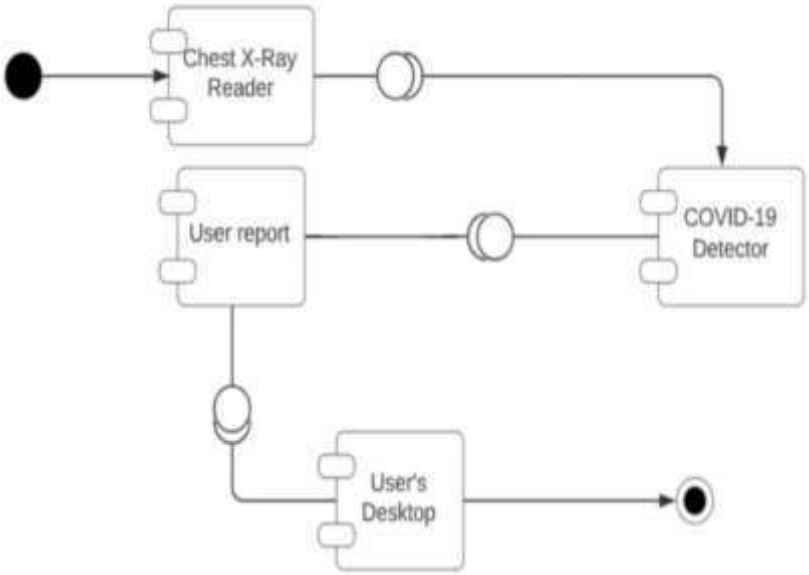


Fig 3.2.8: Component Diagram

Component Diagram shows the in-detail explanation of all the components in the application. Here the

components are Chest X-Ray reader, COVID-19 detector, User Report and User's desktop.

3.2.9 Deployment Diagram

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. Deployment diagrams are typically used to visualize the physical hardware and software of a system.

Deployment diagram shows the configuration of run time processing nodes and the components of the application. It is a kind of structure diagram used in modeling the physical aspects of an object-oriented system.

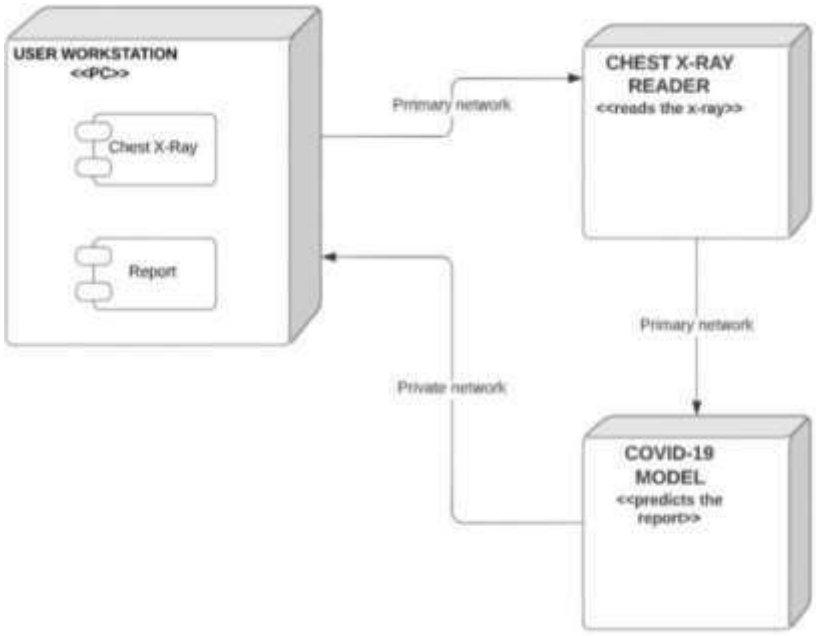


Fig 3.2.9 : Deployment Diagram

Deployment diagram shows the deployment details of the nodes. Here the nodes are User workstation, chest x-ray reader, covid-19 model.

3.2.10 COLLABORATION DIAGRAM

Collaboration diagrams are created by first identifying the structural elements required to carry out the functionality of an interaction. A model is then built using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams.

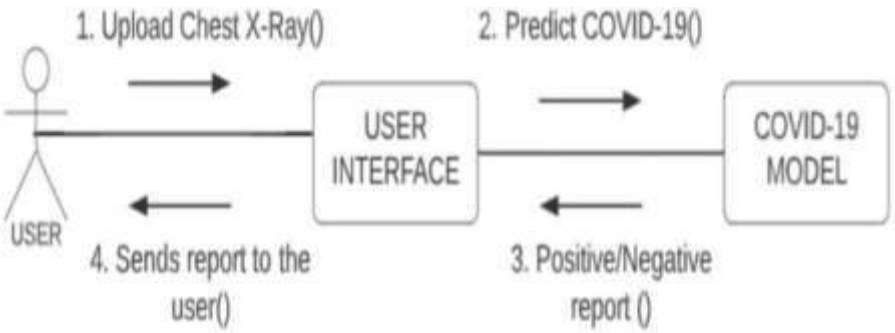


Fig 3.2.10: Collaboration Diagram

Collaboration diagram describes everything about the interactions between the software objects. As shown about the user first uploads the chest x-ray, covid-19 model predicts the presence of virus, send the report to the user.

3.2.11 STATE DIAGRAM

Collaboration diagrams are created by first identifying the structural elements required to carry out the functionality of an interaction. A model is then built using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams.

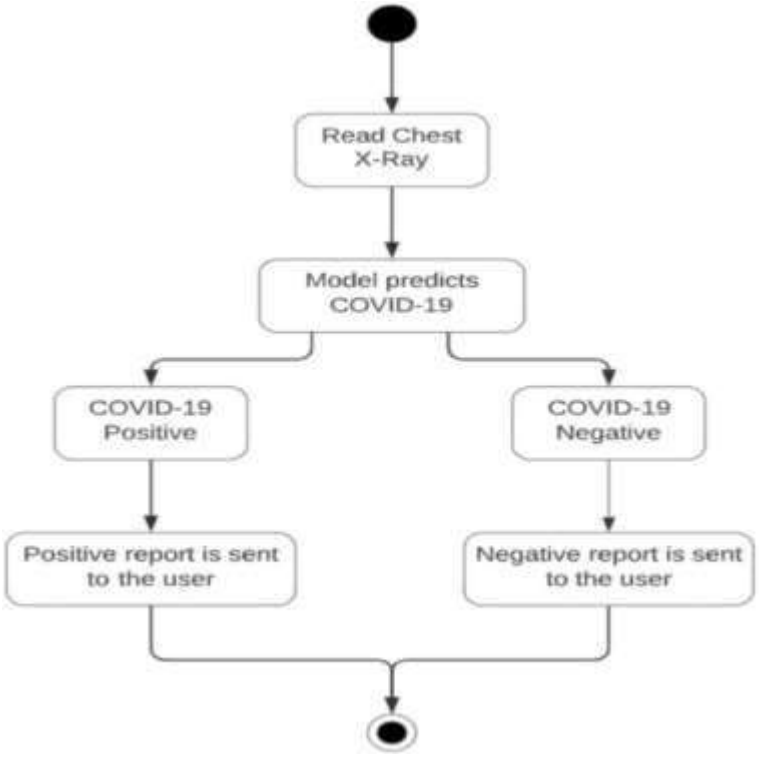


Fig 3.2.11: State Diagram

State chart diagram shows the states of the application in detail as shown above. The flow Reading chest x-ray -> Detection -> Report is shown clearly in the above diagram.

4 Implementation

4.1 Data Preprocessing

Data Pre-processing is the first step of research methodology. In this work, dataset is taken from Kaggle and GitHub. In this step, the input value data is being noise free. This means that error values are segregated from the dataset.

Feature Extraction: The second step is Over-fitting should be avoided as an important objective of feature selection. The performance of model can be improved. This process can be gives the relation between each and every feature of the data with the predestined target data set.

Classification: The CNN classification can be implemented on given dataset. Convolutional neural networks (CNNs) are deep neural networks that have the capability to classify and segment images. CNNs can be trained using supervised or unsupervised machine learning methods, depending on what you want them to do. CNN architectures for classification and segmentation include a variety of different layers with specific purposes, such as a convolutional layer, pooling layer, fully connected layers, dropout layers, etc.

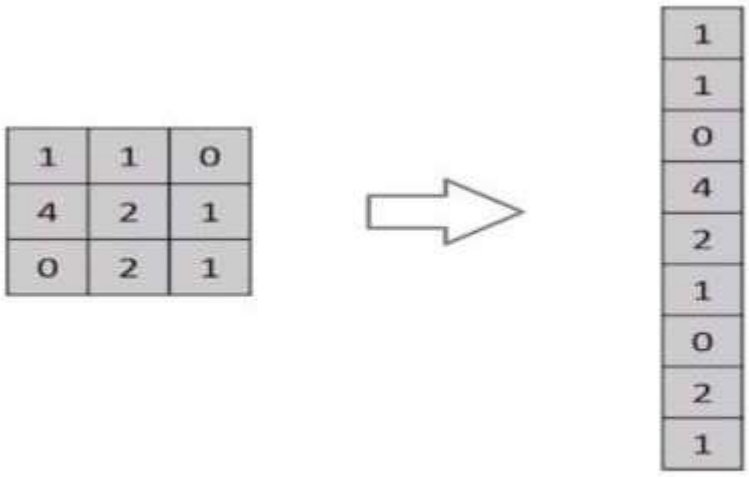
Convolution and max-pooling layers are used for feature extraction. While convolution layers are meant for feature detection, max-pooling layers are meant for feature selection. Max-pooling layers are employed when there are instances when the picture doesn't require all of the high-resolution details or an output with smaller regions extracted by CNN's is needed after performing down sampling operation on input data. The output from convolution and pooling layers is fed into the fully connected layers for classification. The examples of classification learning task where CNN is used are image classification, object detection, and facial recognition.

4.2 Algorithm

A **Convolutional Neural Network (ConvNet/CNN)** is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a ConvNet is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, ConvNets have the ability to learn these filters/characteristics.

The architecture of a ConvNet is analogous to that of the connectivity pattern of Neurons in the Human Brain and was inspired by the organization of the Visual Cortex. Individual neurons respond to stimuli only in a restricted region of the visual field known as the Receptive Field. A collection of such fields overlap to cover the entire visual area.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org



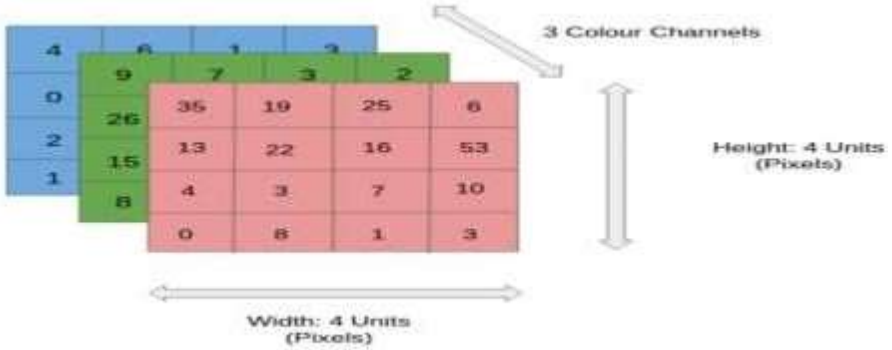
Flattening of a 3×3 image matrix into a 9×1 vector

An image is nothing but a matrix of pixel values, right? So why not just flatten the image (e.g. 3x3 image matrix into a 9x1 vector) and feed it to a Multi-Level Perceptron for classification purposes? Uh.. not really.

In cases of extremely basic binary images, the method might show an average precision score while performing prediction of classes but would have little to no accuracy when it comes to complex images having pixel dependencies throughout.

A ConvNet is able to **successfully capture the Spatial and Temporal dependencies** in an image through the application of relevant filters. The architecture performs a better fitting to the image dataset due to the reduction in the number of parameters involved and reusability of weights. In other words, the network can be trained to understand the sophistication of the image better.

Input Image



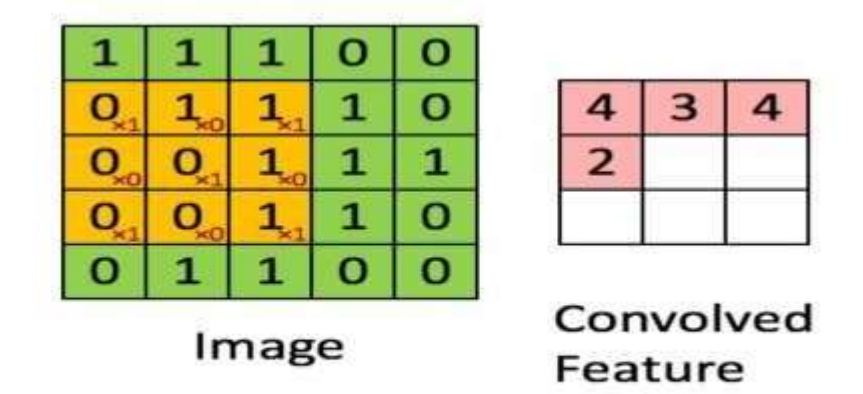
4×4×3 RGB Image

In the figure, we have an RGB image which has been separated by its three colour planes — Red, Green, and Blue. There are a number of such colour spaces in which images exist — Grayscale, RGB, HSV, CMYK, etc.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

You can imagine how computationally intensive things would get once the images reach dimensions, say 8K (7680×4320). The role of the ConvNet is to reduce the images into a form which is easier to process, without losing features which are critical for getting a good prediction. This is important when we are to design an architecture which is not only good at learning features but also is scalable to massive datasets.

Convolution Layer- The Kernel



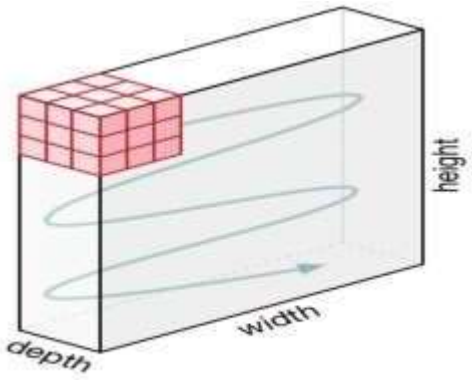
Convoluting a 5×5×1 image with a 3×3×1 kernel to get a 3×3×1 convolved feature

Image Dimensions = 5 (Height) x 5 (Breadth) x 1 (Number of channels, e.g. RGB)

In the above demonstration, the green section resembles our **5x5x1 input image, I**. The element involved in carrying out the convolution operation in the first part of a Convolutional Layer is called the **Kernel/Filter, K**, represented in the colour yellow. We have selected **K as a 3x3x1 matrix.**

Kernel/Filter,	K	=	1	0	1			
			0	1	0			
			1	0	1			

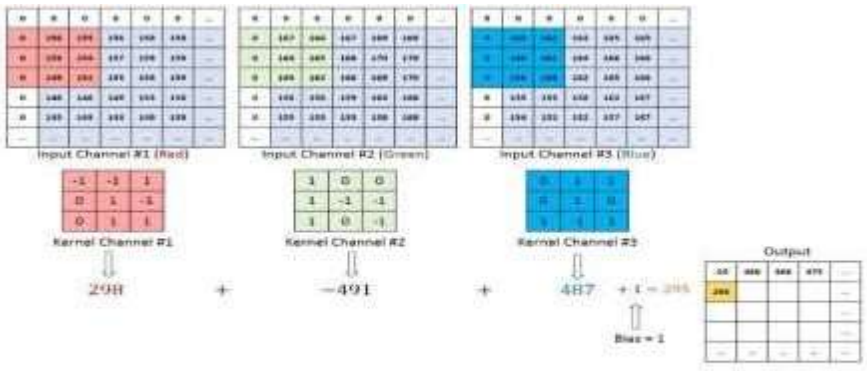
International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org



Movement of the Kernel

The Kernel shifts 9 times because of **Stride Length = 1 (Non-Stride)**, every time performing a **matrix multiplication operation between K and the portion P of the image** over which the kernel is hover in.

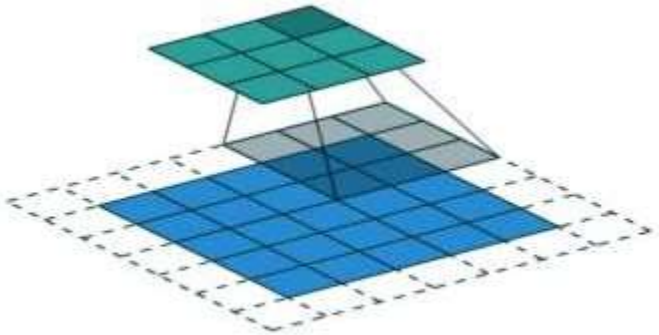
The filter moves to the right with a certain Stride Value till it parses the complete width. Moving on, it hops down to the beginning (left) of the image with the same Stride Value and repeats the process until the entire image is traversed.



Convolution operation on a MxNx3 image matrix with a 3×3×3 Kernel

In the case of images with multiple channels (e.g. RGB), the Kernel has the same depth as that of the input image. Matrix Multiplication is performed between Kn and In stack ([K1, I1]; [K2, I2]; [K3, I3]) and all the results are summed with the bias to give us a squashed one-depth channel Convoluted Feature Output.

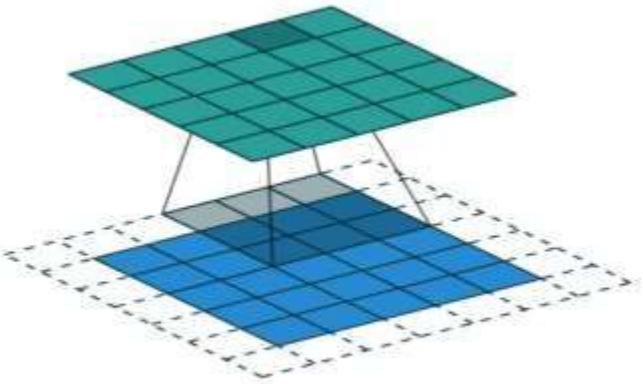
International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org



Convolution Operation with Stride Length = 2

The objective of the Convolution Operation is to **extract the high-level features** such as edges, from the input image. ConvNets need not be limited to only one Convolutional Layer. Conventionally, the first ConvLayer is responsible for capturing the Low-Level features such as edges, colour, gradient orientation, etc. With added layers, the architecture adapts to the High-Level features as well, giving us a network which has the wholesome understanding of images in the dataset, similar to how we would.

There are two types of results to the operation — one in which the convolved feature is reduced in dimensionality as compared to the input, and the other in which the dimensionality is either increased or remains the same. This is done by applying **Valid Padding** in case of the former or same padding in the case of the latter.



SAME padding: 5×5×1 image is padded with 0s to create a 6×6×1 image

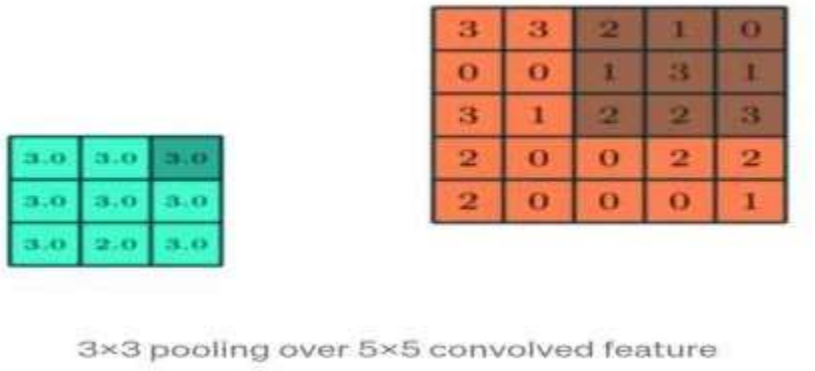
When we augment the 5x5x1 image into a 6x6x1 image and then apply the 3x3x1 kernel over it, we find that the convolved matrix turns out to be of dimensions 5x5x1. Hence the name — **same Padding**.

On the other hand, if we perform the same operation without padding, we are presented with a matrix which has dimensions of the Kernel (3x3x1) itself — **Valid Padding**.

The following repository houses many such GIFs which would help you get a better understanding of how Padding and Stride Length work together to achieve results relevant to our needs.

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Pooling Layer



3×3 pooling over 5×5 convolved feature

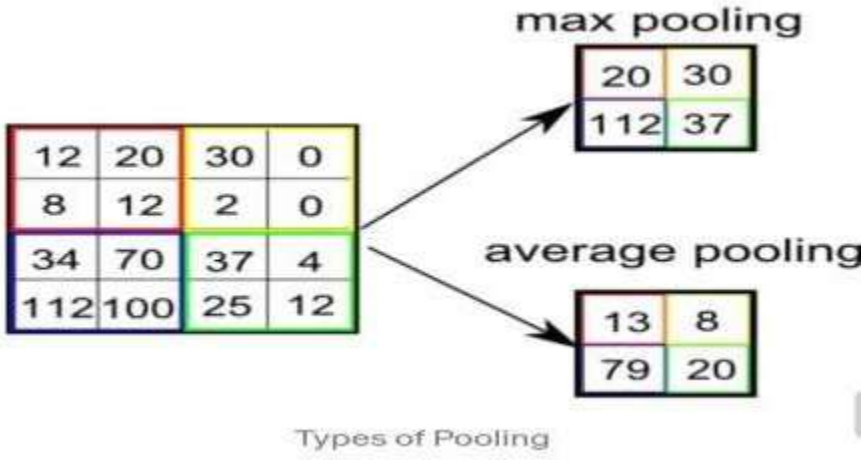
Similar to the Convolutional Layer, the Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to **decrease the computational power required to process the data** through dimensionality reduction. Furthermore, it is useful for **extracting dominant features** which are rotational and positional invariant, thus maintaining the process of effectively training of the model.

There are two types of Pooling: Max Pooling and Average Pooling. **Max Pooling** returns the **maximum value** from the portion of the image covered by the Kernel. On the other hand, **Average Pooling** returns the **average of all the values** from the portion of the image covered by the Kernel.

Max Pooling also performs as a **Noise Suppressant**. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction. On the other hand, Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, we can say that **Max Pooling performs a lot better than**

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

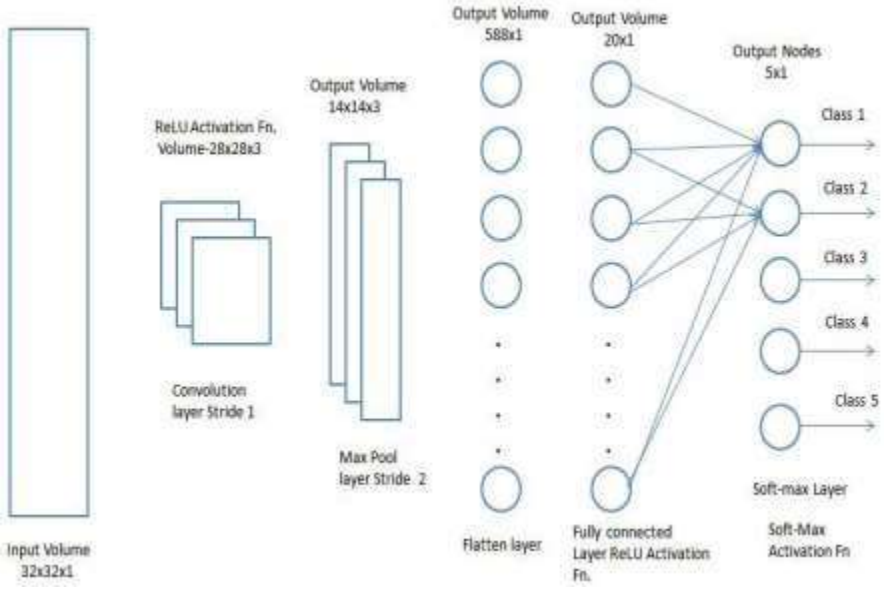
Average Pooling.



Types of Pooling

The Convolutional Layer and the Pooling Layer, together form the i-th layer of a Convolutional Neural Network. Depending on the complexities in the images, the number of such layers may be increased for capturing low-levels details even further, but at the cost of more computational power.

After going through the above process, we have successfully enabled the model to understand the features. Moving on, we are going to flatten the final output and feed it to a regular Neural Network for classification purposes.



Classification — Fully Connected Layer (FC Layer)

Adding a Fully-Connected layer is a (usually) cheap way of learning non-linear combinations of the high-level features as represented by the output of the convolutional layer. The Fully-Connected layer is learning a possibly non-linear function in that space.

Now that we have converted our input image into a suitable form for our Multi-Level Perceptron, we shall flatten the image into a column vector. The flattened output is fed to a feed-forward neural network and back propagation applied to every iteration of training. Over a series of epochs, the model is able to distinguish between dominating and certain low-level features in images and classify them using the **Softmax Classification** technique.

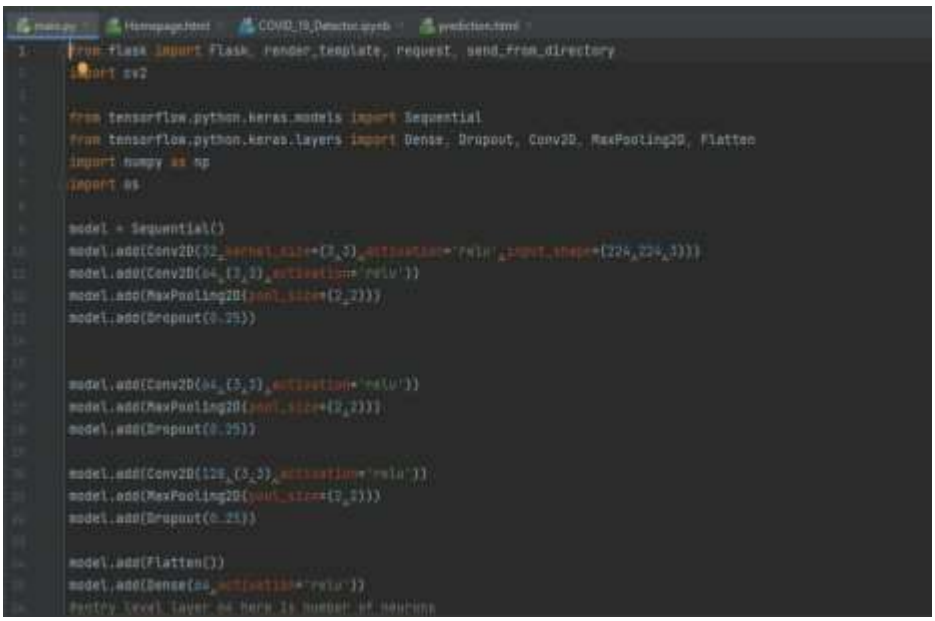
4.3 Modules

The application provides the following user interfaces:

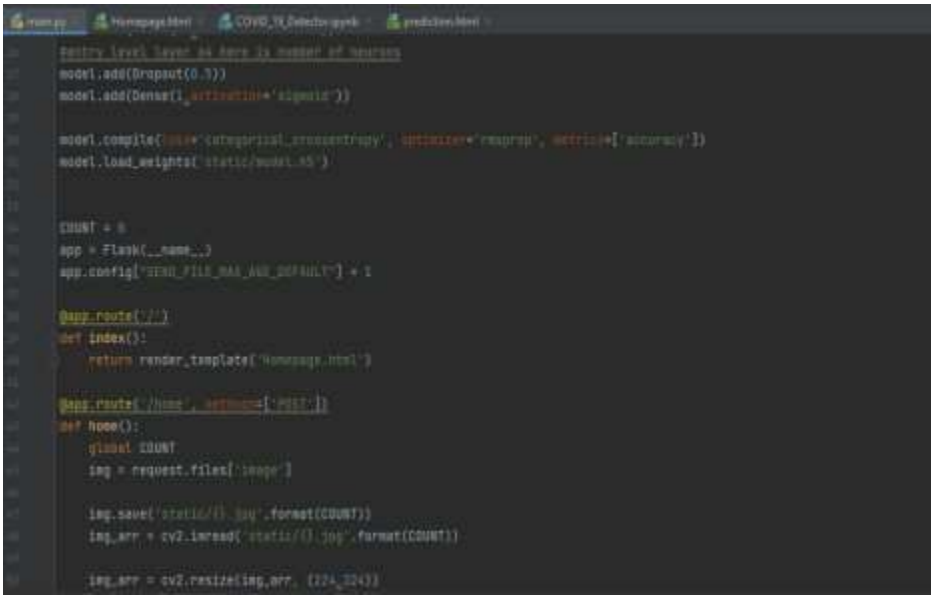
Covid-19 Detection: This section gives the user an option to upload his/her chest x-ray and by clicking the predict button the report of whether he/she is tested COVID-19 positive or negative is generated and displayed.

General Awareness: In this section the user can get all the information about the COVID-19 symptoms, precautions to be taken, vaccine information etc.

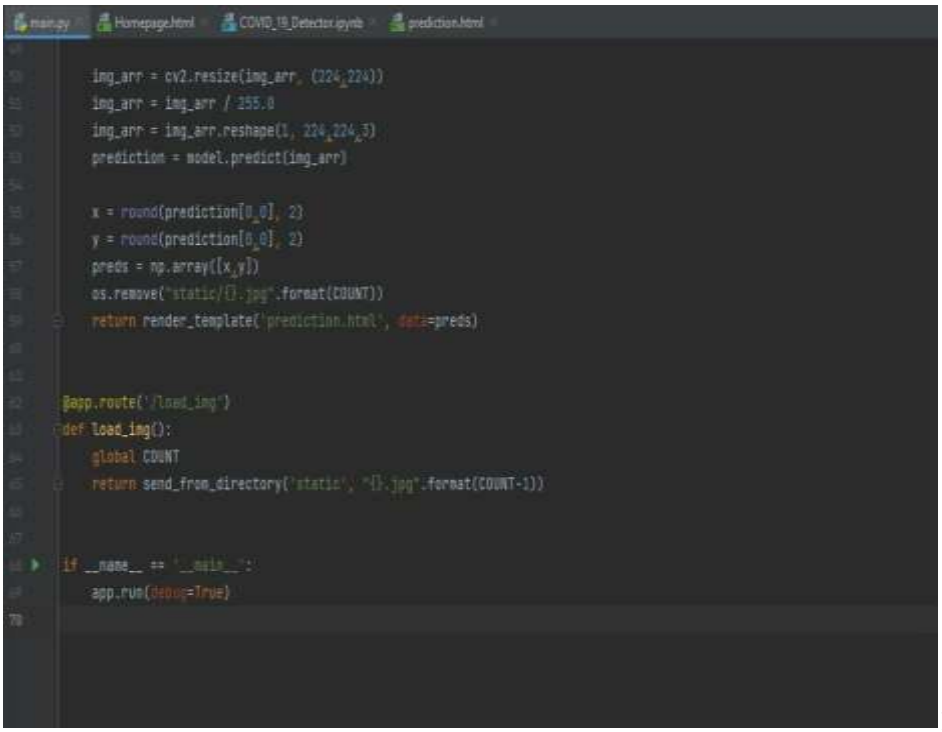
4.4 Code Implementation



International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

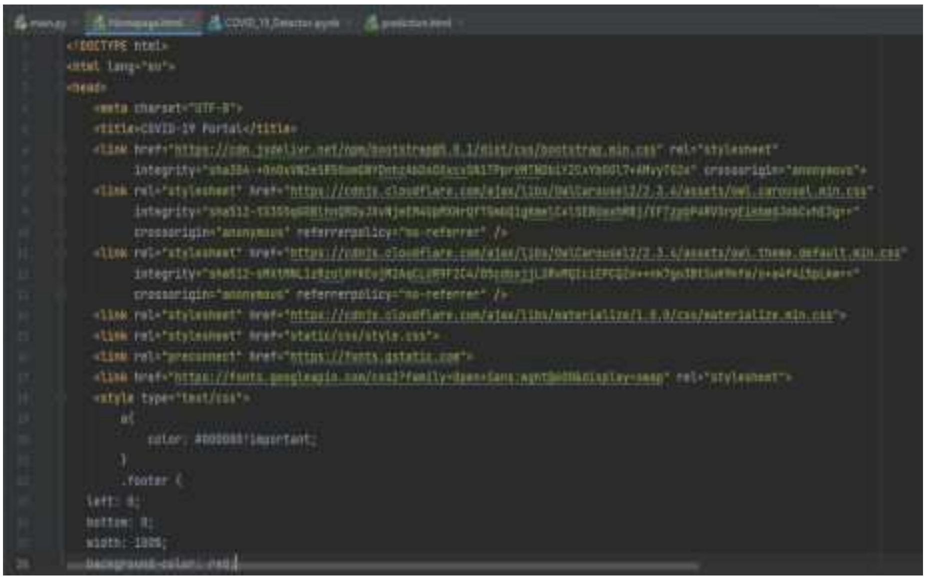


```
model.add(Dropout(0.5))
model.add(Dense(1,activation='sigmoid'))

model.compile(loss='categorical_crossentropy', optimizer='rmsprop', metrics=['accuracy'])
model.load_weights('static/model.h5')


COUNT = 0
app = Flask(__name__)

@app.route('/')
def index():
    return render_template('Homepage.html')

@app.route('/home', methods=['POST'])
def home():
    global COUNT
    img = request.files['image']

    img.save('static/{}.jpg'.format(COUNT))
    img_arr = cv2.imread('static/{}.jpg'.format(COUNT))

    img_arr = cv2.resize(img_arr, (224,224))
```



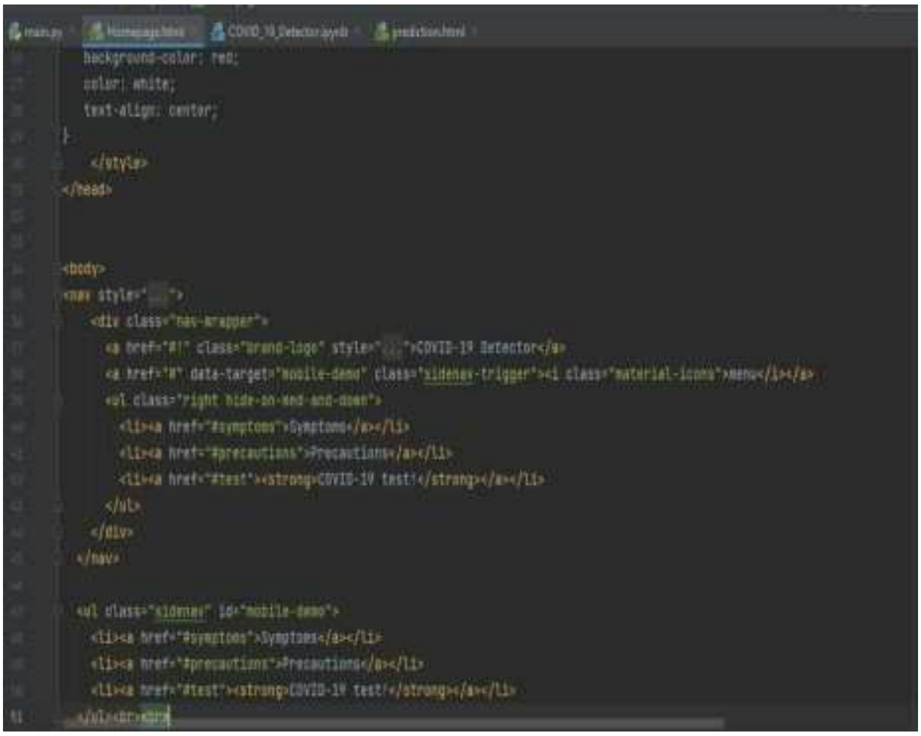
```
    img_arr = cv2.resize(img_arr, (224,224))
    img_arr = img_arr / 255.0
    img_arr = img_arr.reshape(1, 224,224,3)
    prediction = model.predict(img_arr)

    x = round(prediction[0,0], 2)
    y = round(prediction[0,0], 2)
    preds = np.array([x,y])
    os.remove("static/{}.jpg".format(COUNT))
    return render_template('prediction.html', data=preds)


@app.route('/load_img')
def load_img():
    global COUNT
    return send_from_directory('static', "{}.jpg".format(COUNT-1))


if __name__ == '__main__':
    app.run(debug=True)
```

Fig 4.2.1: Main.py



main.py | Homepage.html | COVID_19_Detector.ipynb | prediction.html

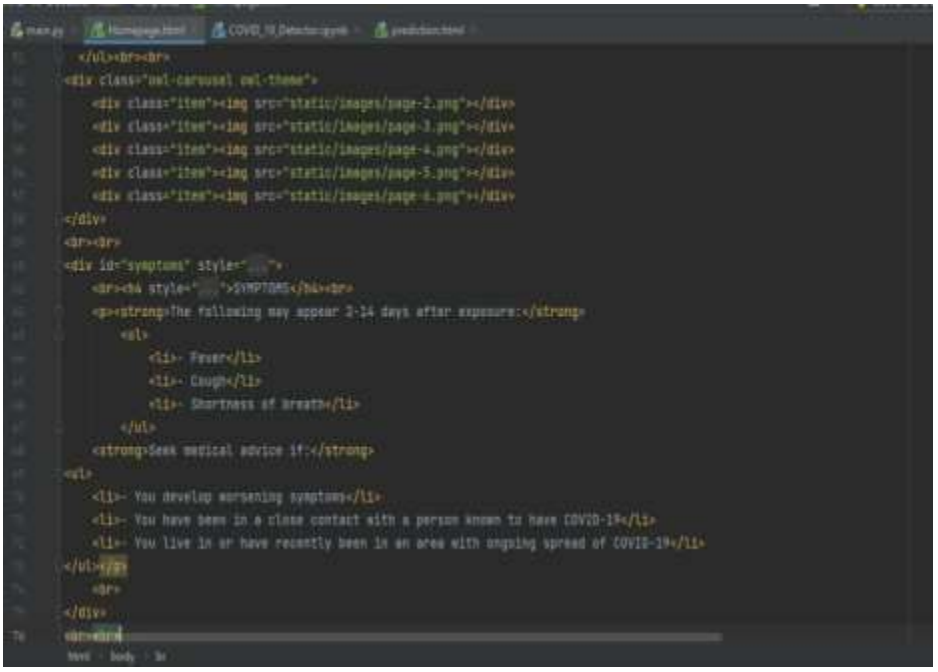
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>COVID-19 Portal</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.1/dist/css/bootstrap.min.css" rel="stylesheet"
        integrity="sha384-[illegible]" crossorigin="anonymous">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/OwlCarousel2/2.3.4/assets/owl.carousel.min.css"
        integrity="sha512-[illegible]"
        crossorigin="anonymous" referrerpolicy="no-referrer" />
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/OwlCarousel2/2.3.4/assets/owl.theme.default.min.css"
        integrity="sha512-[illegible]"
        crossorigin="anonymous" referrerpolicy="no-referrer" />
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/materialize/1.0.0/css/materialize.min.css">
    <link rel="stylesheet" href="static/css/style.css">
    <link rel="preconnect" href="https://fonts.gstatic.com">
    <link href="https://fonts.googleapis.com/css2?family=Open+Sans:wght@600&display=swap" rel="stylesheet">
    <style type="text/css">
        a{
            color: #000000!important;
        }
        .footer {
  left: 0;
  bottom: 0;
  width: 100%;
  background-color: red;
```



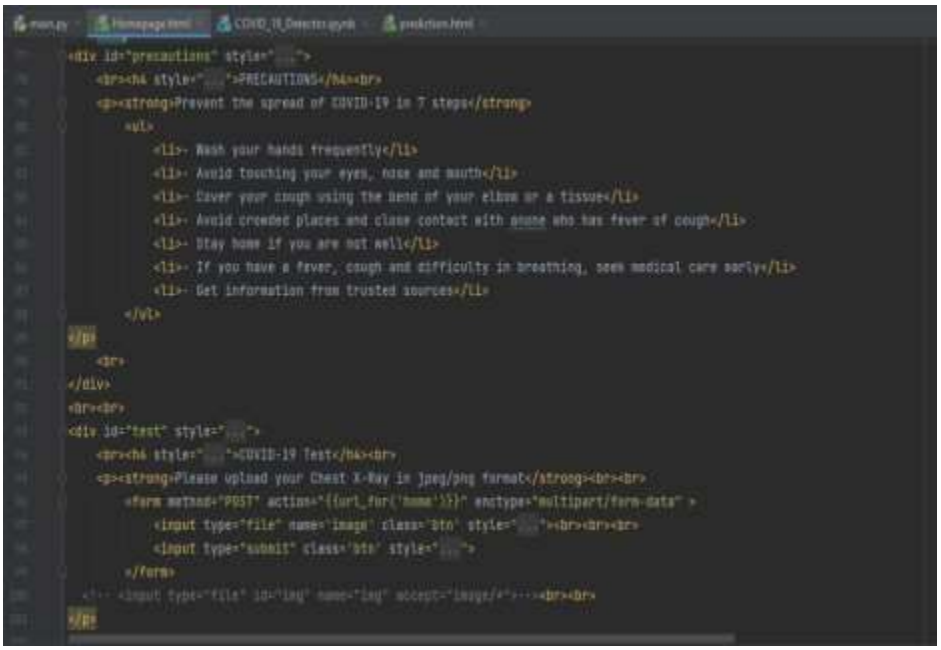
main.py | Homepage.html | COVID_19_Detector.ipynb | prediction.html

```
  background-color: red;
  color: white;
  text-align: center;
}
    </style>
</head>


<body>
<nav style="...">
    <div class="nav-wrapper">
      <a href="#!" class="brand-logo" style="...">COVID-19 Detector</a>
      <a href="#" data-target="mobile-demo" class="sidenav-trigger"><i class="material-icons">menu</i></a>
      <ul class="right hide-on-med-and-down">
        <li><a href="#symptoms">Symptoms</a></li>
        <li><a href="#precautions">Precautions</a></li>
        <li><a href="#test"><strong>COVID-19 test!</strong></a></li>
      </ul>
    </div>
  </nav>

  <ul class="sidenav" id="mobile-demo">
    <li><a href="#symptoms">Symptoms</a></li>
    <li><a href="#precautions">Precautions</a></li>
    <li><a href="#test"><strong>COVID-19 test!</strong></a></li>
  </ul><br><br>
```

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org



```
</ul><br><br>
<div class="owl-carousel owl-theme">
    <div class="item"><img src="static/images/page-2.png"></div>
    <div class="item"><img src="static/images/page-3.png"></div>
    <div class="item"><img src="static/images/page-4.png"></div>
    <div class="item"><img src="static/images/page-5.png"></div>
    <div class="item"><img src="static/images/page-6.png"></div>
</div>
<br><br>
<div id="symptoms" style="...">
    <br><h4 style="...">SYMPTOMS</h4><br>
    <p><strong>The following may appear 2-14 days after exposure:</strong>
        <ul>
            <li>- Fever</li>
            <li>- Cough</li>
            <li>- Shortness of breath</li>
        </ul>
    <strong>Seek medical advice if:</strong>
<ul>
    <li>- You develop worsening symptoms</li>
    <li>- You have been in a close contact with a person known to have COVID-19</li>
    <li>- You live in or have recently been in an area with ongoing spread of COVID-19</li>
</ul></p>
    <br>
</div>
<br><br>
```



```
<div id="precautions" style="...">
    <br><h4 style="...">PRECAUTIONS</h4><br>
    <p><strong>Prevent the spread of COVID-19 in 7 steps</strong>
        <ul>
            <li>- Wash your hands frequently</li>
            <li>- Avoid touching your eyes, nose and mouth</li>
            <li>- Cover your cough using the bend of your elbow or a tissue</li>
            <li>- Avoid crowded places and close contact with anyone who has fever of cough</li>
            <li>- Stay home if you are not well</li>
            <li>- If you have a fever, cough and difficulty in breathing, seek medical care early</li>
            <li>- Get information from trusted sources</li>
        </ul>
</p>
    <br>
</div>
<br><br>
<div id="test" style="...">
    <br><h4 style="...">COVID-19 Test</h4><br>
    <p><strong>Please upload your Chest X-Ray in jpeg/png format</strong><br><br>
        <form method="POST" action="{{url_for('home')}}" enctype="multipart/form-data" >
            <input type="file" name="image" class="btn" style="..."><br><br><br>
            <input type="submit" class="btn" style="...">
        </form>
    <!-- <input type="file" id="img" name="img" accept="image/*"><br><br>
</p>
```

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

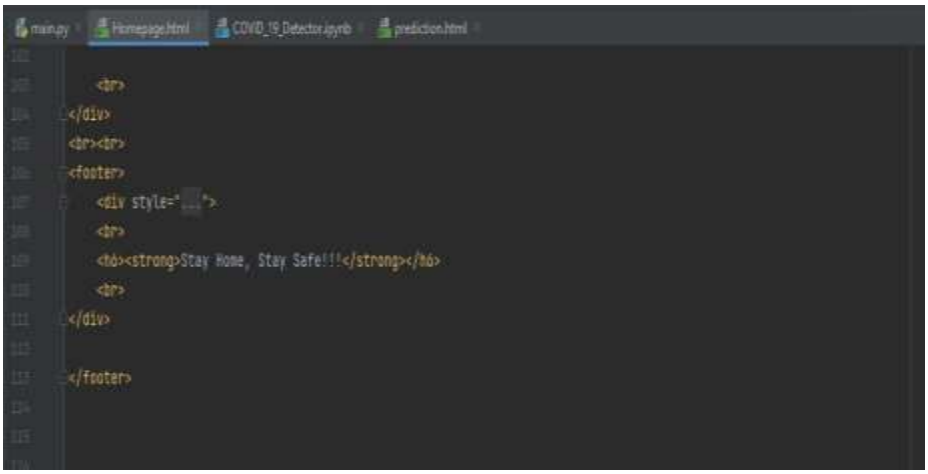
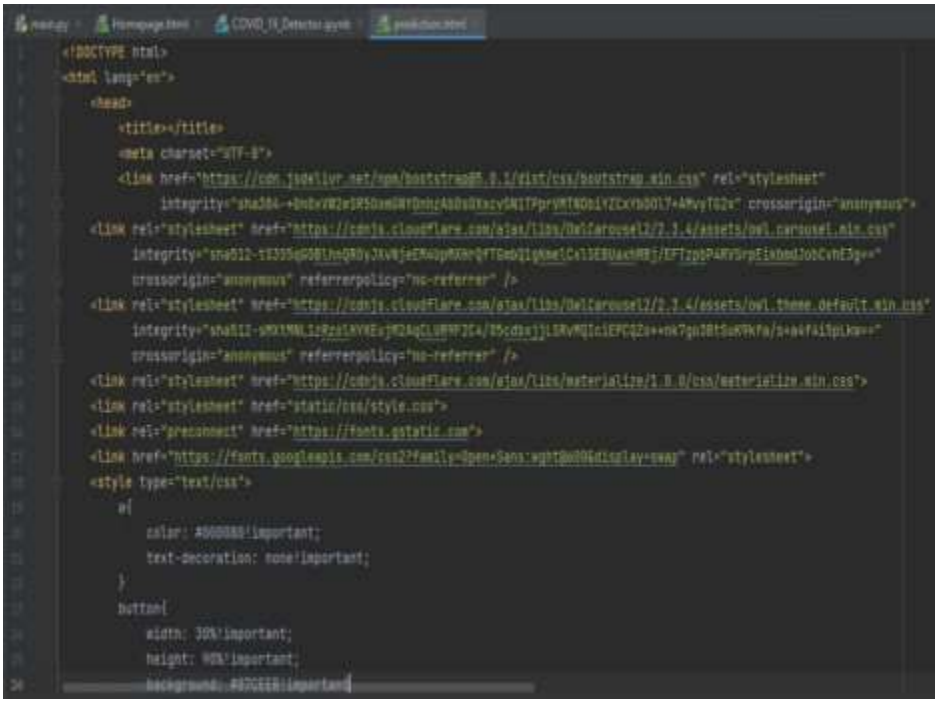
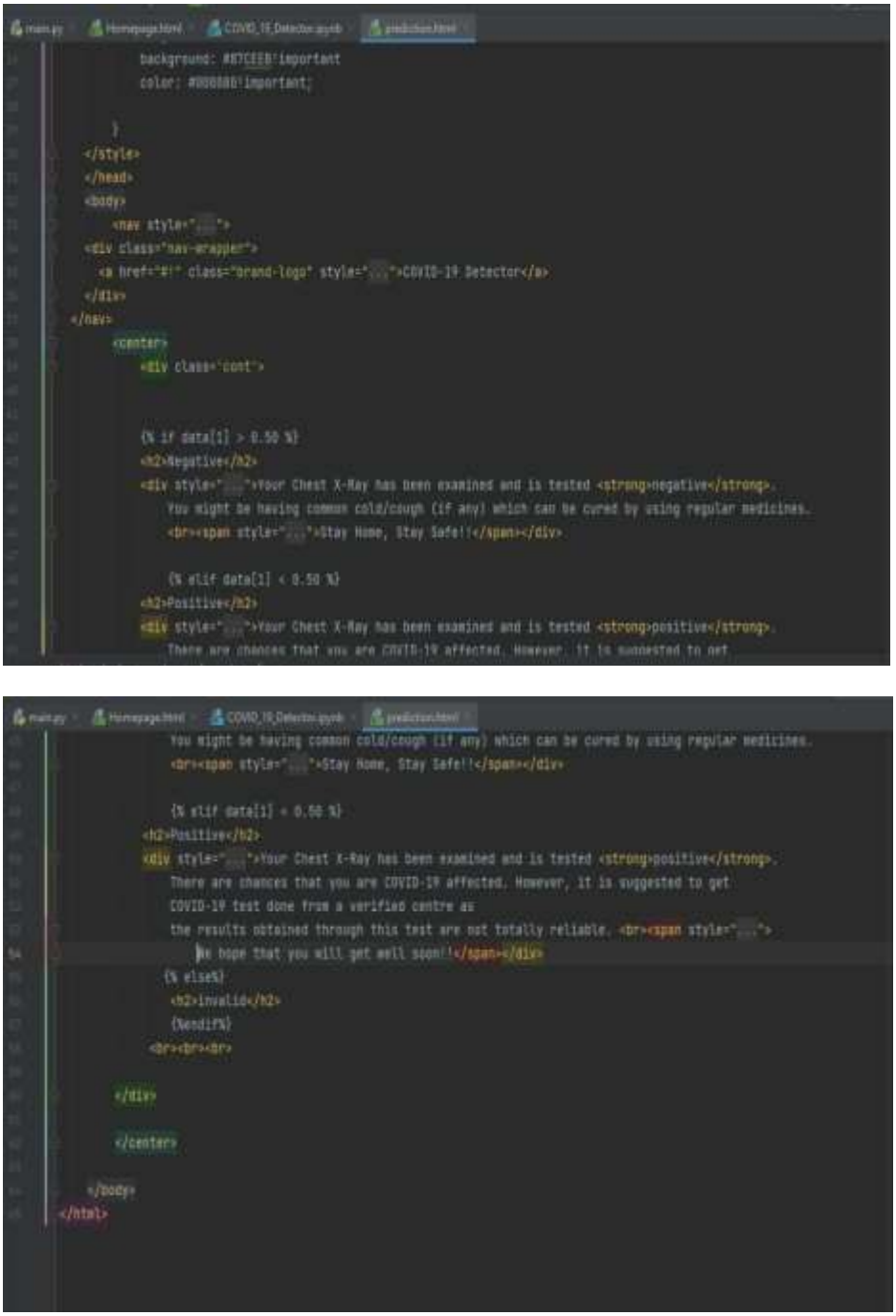


Fig 4.2.2: Homepage.html



International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org



```
            background: #B7CEEB!important
            color: #000080!important;

        }
    </style>
    </head>
    <body>
        <nav style="...">
    <div class="nav-wrapper">
      <a href="#!" class="brand-logo" style="...">COVID-19 Detector</a>
    </div>
    </nav>
        <center>
            <div class='cont'>


            {% if data[1] > 0.50 %}
            <h2>Negative</h2>
            <div style="...">Your Chest X-Ray has been examined and is tested <strong>negative</strong>.
                You might be having common cold/cough (if any) which can be cured by using regular medicines.
                <br><span style="...">Stay Home, Stay Safe!!</span></div>

                {% elif data[1] < 0.50 %}
            <h2>Positive</h2>
            <div style="...">Your Chest X-Ray has been examined and is tested <strong>positive</strong>.
                There are chances that you are COVID-19 affected. However, it is suggested to get
                COVID-19 test done from a verified centre as
                the results obtained through this test are not totally reliable. <br><span style="...">
                   We hope that you will get well soon!!</span></div>
               {% else%}
                <h2>Invalid</h2>
                {%endif%}
             <br><br><br>

        </div>

        </center>

    </body>
</html>
```

Fig 4.2.3: Prediction.html

```
TRAIN_PATH = "covidDataset/Train"
VAL_PATH="CovidDtaset/Test"
```

```
import numpy as np
import matplotlib.pyplot as plt
import keras
from keras.layers import *
from keras.models import *
from keras.preprocessing import image
```

```
#CNN Based Model in Keras
model = Sequential()
model.add(Conv2D(32,kernel_size=(3,3),activation='relu',input_shape=(224,224,3)))
model.add(Conv2D(64,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))

model.add(Conv2D(64,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))

model.add(Conv2D(128,(3,3),activation='relu'))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Dropout(0.25))

model.add(Flatten())
model.add(Dense(64,activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(1,activation='sigmoid'))

model.compile(loss=keras.losses.binary_crossentropy,optimizer='adam',metrics=['accuracy'])
```

```
model.summary()
```

```
Model: "sequential_6"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_14 (Conv2D)           (None, 222, 222, 32)      896

conv2d_15 (Conv2D)           (None, 220, 220, 64)      18496

max_pooling2d_9 (MaxPooling  (None, 110, 110, 64)      0
2D)

dropout_12 (Dropout)         (None, 110, 110, 64)      0

conv2d_16 (Conv2D)           (None, 108, 108, 64)      36928
```

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

```
max_pooling2d_10 (MaxPoolin  (None, 54, 54, 64)       0
g2D)

dropout_13 (Dropout)         (None, 54, 54, 64)       0

conv2d_17 (Conv2D)           (None, 52, 52, 128)      73856

max_pooling2d_11 (MaxPoolin  (None, 26, 26, 128)      0
g2D)

dropout_14 (Dropout)         (None, 26, 26, 128)      0

flatten_3 (Flatten)          (None, 86528)            0

dense_6 (Dense)              (None, 64)               5537856

dropout_15 (Dropout)         (None, 64)               0

dense_7 (Dense)              (None, 1)                65

=================================================================
Total params: 5,668,097
Trainable params: 5,668,097
Non-trainable params: 0
_________________________________________________________________
```

```
#train from scarch
train_datagen=image.ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
)
test_dataset=image.ImageDataGenerator(rescale=1./255)

train_generator=train_datagen.flow_from_directory(
   'CovidDataset/Train' ,
   target_size=(224,224),
   batch_size=32,
   class_mode='binary'
)
```

```
Found 224 images belonging to 2 classes.
```

```
train_generator.class_indices
```

```
{'Covid': 0, 'Normal': 1}
```

```
validation_generator=test_dataset.flow_from_directory(
    'CovidDataset/Val',
    target_size=(224,224),
    batch_size=32,
```

Fig 4.2.4: Covid-Detector (CNN Code)

4.5 TESTING

The main aim of this system testing process was to determine all defects in our project. the program was subjected to a set of test inputs and various observations were made and based on these observations it will be decided whether the program behaves as expected or not. This project consists of program level testing, modules level testing integrated and carried out.

WHITE BOX TESTING: white box testing sometimes called as "glass box testing" is a test case design uses the control structure of the procedural design to drive test case. Using white box testing methods ,the tests were made on the system:

A) All independent paths within a module have been exercised once.

B) All logical decisions were checked for the truth and falsity of the values.

BLACK BOX TESTING: Black box testing focuses on the functional requirements of the software. This is black box testing enables the software engineering to derive a set of input conditions that will fully exercise all functional requirements for a program.

1) Interface errors

2) Performance in data structure

3) Performance errors

4) Initializing and termination errors

4.5.1 TEST CASES

S.NO	Test Case desc	Expected Result	Actual Result	Priority
1	Click on submit button without uploading an image.	Should show error	Showing error	High
2	Upload a report which is proven for negative	Should show negative report	Negative report	High
3	Upload a report which is proven for positive	Should show positive report	Positive report	High

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

4	Click on submit	To navigate to next page	Navigating to next page	High
5	Click on Symptoms	To navigate to symptoms page	Navigating to symptoms page	High
6	Click on Precautions	To navigate to precautions page	Navigating to precautions page	High
7	Click on Covid-19 test page	To navigate to covid-19 test page	Navigating to Covid-19 test page	High

5 RESULTS

Fig 5.1: Home Page-i

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

Fig 5.2: Home Page-ii

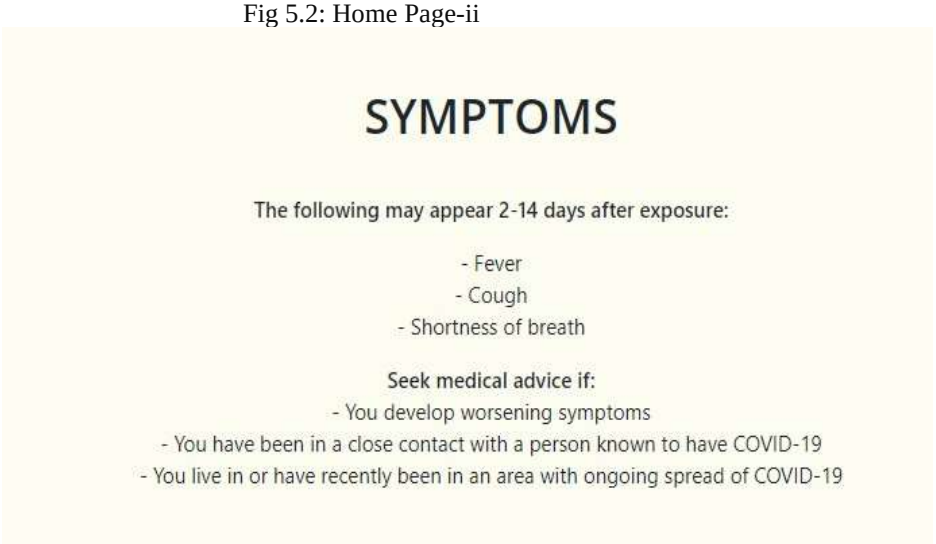


Fig 5.3: Symptoms Page

International Journal for Innovative Engineering and Management Research
A Peer Revieved Open Access International Journal
www.ijiemr.org

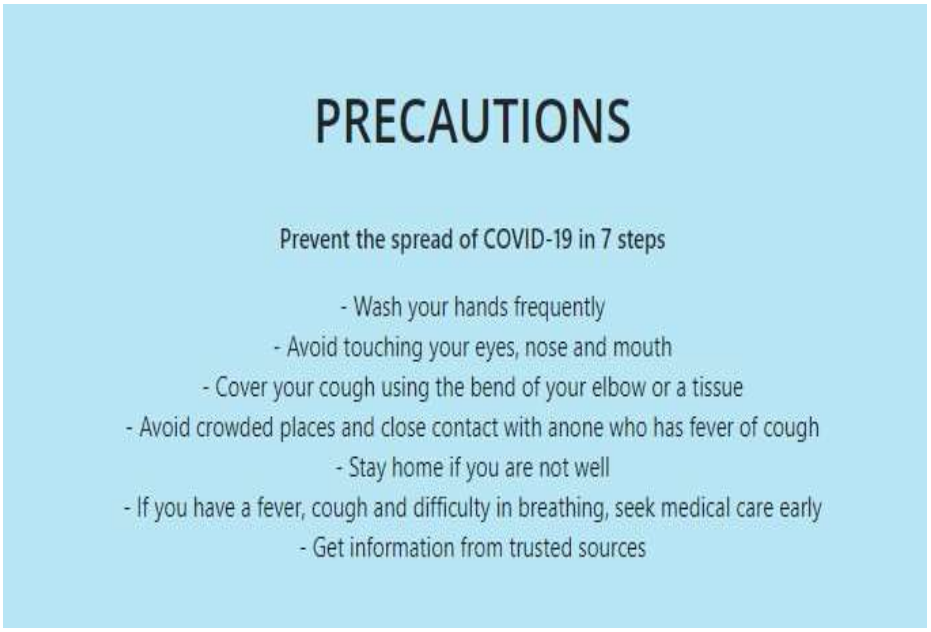


Fig 5.4: Precautions Page

Fig 5.5: Covid-19 Test Page

Fig 5.6: Positive Report

Fig 5.7: Negative Report

6 Conclusion

The number of people infected with COVID-19 has risen rapidly. Machine vision techniques and artificial intelligence are critical in diagnosing and treating disease. The purpose of this project was to propose a method for the "COVID-19" problem via a set of lung images that included two categories of COVID-19, and healthy.

A deep convolutional neural network consisting of 11 layers was applied to extract the features. The binary differential metaheuristic method was used to select relevant features and eliminate unrelated features. Lung X-ray images were classified using an SVM classifier based on these optimal features. This study demonstrated that the accuracy indicator and the number of relevant extracted features outperformed previous methods using the same data. Based on a deep neural network and a metaheuristic feature selection algorithm, the proposed model can be used in various other medical applications.

7 Future Scope

It is not possible to develop a system that makes all the requirements of the user. User requirements keep changing as the system is being used. Some of the future enhancements that can be done to this system is:

- As the technology emerges, it is possible to upgrade the system and can be adaptable to accept any view of an X-ray.
- As of now we getting 98% accuracy in future we will try improving it. Based on the future security issues, security can be improved using emerging technologies like single sign-on.
- We will also try to enhance the interface so that it looks more attractive and more interactive.
- We will automate our application.
- We will be creating a mobile application for this project.

8 References

[1] Julfikar Haider, Marcin Kowalski, Md. Abdul Based, Mominul Ahsan, Nur-A-Alam, "COVID-19 Detection from Chest X-ray Images Using Feature Fusion and Deep Learning", Sensors, 2021.

[2] Asif Iqbal Khan, Junaid Latief Shah, Mohammad Mudasir Bhat, "CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images", Elsevier B.V, 2020.

[3] Kamal Kumar Ghanshala, Rahul Chauhan, R.C Joshi, "Convolutional Neural Network (CNN) for Image Detection and Recognition" ,Graphic Era University, Dehradun, December 2018.

[4] Chung J.H., Elicker B.M., Kanne J.P., Ketai L.H., Little B.P," Essentials for radiologists on COVID-19: An update—Radiology scientific expert panel" RSNA, May 2020.

[5] Saad Albawi, Saad Al-Zawi, Tareq Abed Mohammed, "Understanding of a convolutional neural network", Department of Computer Engineering, Istanbul Kemerburgaz University, Istanbul, Turkey, August 2017.

[6] DhwaaniParikh; Vineet Menon, "Machine Learning Applied to Cervical Cancer Data", Association of Modern Education and Computer Science, January 2019, India.

[7] Luca Abel, "Classification and Prediction of Acute Stress-Induced Response Patterns", Bachelor's Thesis in Medical Engineering, January 2019, Malsch.

[8] Ankush Mittal, DimpyVarshni, Kartik Thakral, Lucky Agarwal, Rahul Nijhawan, "Pneumonia Detection Us- ing CNN based Feature Extraction", College of Engineering Roorkee, Graphic Era University Dehradun, August2020, India.

[9] AyishaShamna.KK, Jamsheera. K, Shameena.P, "CNN Based Landmark Detection and Alzheimer's Diagnosis Using Landmark Feature", Cochin College of Engineering and Technology, Velachery, Kerala, February 2018, India

Anisha P R , Kishor Kumar Reddy C and Nguyen Gia Nhu, "Blockchain Technology: A Boon at the Pandemic Times – A Solution for Global Economy Upliftment with AI and IoT", EAI/Springer Innovations in Communication and Computing, 2022.

PR Anisha, CKK Reddy, NG Nhu, Blockchain Technology: A Boon at the Pandemic Times–A Solution for Global Economy Upliftment with AI and IoT, Blockchain Security in Cloud Computing, 227-252, 2022

PR Anisha, Kishor Kumar Reddy C, NG Nguyen, G Sreelatha, A Text Mining using Web Scraping for Meaningful Insights, Journal of Physics: Conference Series 2089 (1), 012048, 2021

CKK Reddy, PR Anisha, RM Mohana, Assessing Wear Out of Tyre using Opencv & Convolutional Neural Networks, Journal of Physics: Conference Series 2089 (1), 012001, 2021