Project Report on

" *Pneumonia Detection using Deep Learning* "

BACHELOR OF ENGINEERING
in
INFORMATION SCIENCE AND ENGINEERING
by

Name

A GAUTHAM

Under the Guidance of

Mr. CHANDRASHEKHARA K T

BE M Tech, (PhD) , Associate Professor
DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING, BMSIT&M

BMS INSTITUTE OF TECHNOLOGY& MANAGEMENT
BENGALURU-560064

2021-22

BMS INSTITUTE OF TECHNOLOGY& MANAGEMENT

YELAHANKA, BENGALURU-560064

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project based learning entitled " ***Pneumonia Detection using Deep Learning*** **"** is a bonafide work carried out by **Mr. A GAUTHAM**

during the academic year 2021-22. It is certified that all corrections/suggestions indicated for the Internal Assessment have been incorporated in the report deposited in the departmental library.

Signature of the guide	Signature of the Coordinator	Signature of the HOD
Prof. Chandrashekhara K.T	Prof. Shwetha M.S	Dr.Pushpa S.K

ABSTRACT

- The outbreak of the novel Corona virus disease (COVID-19) in December 2019 has led to global crisis around the world. The disease was declared pandemic by World Health Organization (WHO) on 11th of March 2020.
- (RT-PCR) is the standard method for detection of COVID-19 disease, but it has many challenges such as false positives, low sensitivity, expensive, and requires experts to conduct the test.
- Chest X-ray (CXR) scan images can be considered as an alternative as they are fast to obtain and easily accessible.
- There are number of approaches to classify CXR images and detect the COVID-19 infections, the majority of these approaches can only recognize two classes (e.g., COVID-19 vs. normal)
- The current work proposes the use of a deep learning approach based on trained Keras models.
- This model performs two-way classification ,three-way classification and four-way classification.

ACKNOWLEDGEMENT

We have put in a lot of effects into this project. However, it would not have been possible without the help and support of a lot of individuals. We would like to extend our sincere thanks to all of them. We would like to extend our gratitude to our guide, Prof. Chandrashekhara K.T for his supervision as well as providing necessary information regarding the project.

We would like to extend our gratitude to our professors and friends for giving their time to help us.

Thank you.

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

1.1 PREAMBLE

- In this Project we aim to dive into a Present Societal Pandemic issue which we are facing around us past 2 years due to outbreak of Novel Corona Virus.

- Getting tested for covid-19 virus is not an easy deal with costly RT-PCR test, and delayed results, and with its no. of variants with different mutations emerging everyday all the new methods found to detect the virus and its variant have either become :
  - Ineffective as all tests may not find each of the variants.
  - Each of them a set a finical restrictions for the technology used.
  - Each test has its own detection time .
- Chest X-Ray already exists and overcomes most of the above drawbacks, but still fail to give long term effects or severity.
- So as a Solution We Aim to develop a model to give large no. of classifications and comparisons of Covid Patients and whether it leads to pneumonia disease , also these models could be trained to classify long term effects after years of infection how things could change w.r.t chest infections, and lead to other chronic disease.

1.2 MOTIVATION

- In December 2019, Wuhan a city in China was affected by a deadly, gruesome Pneumonia which was declared a pandemic by the World Health Organization. But the reason for the outbreak was not clear to everyone. Later, the doctors identified the disease as a new species of coronavirus, also currently known as COVID-19.
- The main motivation behind this project is to identify Pneumonia just by using the X-Ray images of the patients.
- As doctors must do a lot of certain tests to identify if the patient has Pneumonia or not. To solve this cumbersome problem, an ensemble of deep learning models is developed to make the work of doctors easier.
- This model could help mitigate the reliability and interpretability challenges often faced when dealing with medical imagery.

1.3 OBJECTIVES

The objectives of our project are as follows :

- Successful detection of presence of pus/fluid filled air sacs in the lungs which indicates pneumonia.
- Differentiate b/w various types of pneumonia like bacterial pneumonia, non-Covid-19 viral pneumonia and Covid-19 pneumonia.
- Increase the accuracy and efficiency of the model to detect pneumonia.

1.4 PROBLEM STATEMENET

- To detect Pneumonia in patients with covid, Classify them on the basis of severity and compare them with various other covid and non-covid Pneumonia. To study their long term effects even after patient recovering from Covid, but may pertain Pneumonia or related chronic Disease.

CHAPTER 2

2.1 EXISTING SYSTEM

- During the surge of Covid especially during second wave , which dealt with breathing issues and chest infections, and X-ray turned be a vital tool to classify in covid positive patients.
- The doctors with immense workload, and lot of mental pressure would have to classify and determine results of X-rays with naked eyes which is tedious, but to at least cross verify, we can use deep learning models which is turning out be great with the good accuracy and high speed of results speed of results.

2.2 PROPOSED SYSTEM

- The existing system can detect pneumonia in normal patients without covid, but this system ensures to detect pneumonia in the patients suffering from covid with greater accuracy and efficacy using Chest X-ray images.

2.3 TOOLS USED

❖ HARDWARE

- Intel Core 5 and above (recommended)
- Quad processor or higher
- CPU cores – 4
- RAM - 8 GB Min
- No of parallel threads - 4
- 8 GB of available disk space minimum
- Cache - 8 mb
- Freq - 1066 Hz
- Clock speed - 3 GHz
- 1200 x 800 min screen resolution

❖ SOFTWARE

- Programming Language – Python
- Operating System – Windows 8/10 64-bit , Ubuntu
- Software tool used – Anaconda, Jupyter Notebook, Spyder.

2.4 RELEVANCE TO SOCIETY / INDUSTRY

- ❖ Pneumonia is one of the most severe, life-threatening and contagious disease
- ❖ We, in this model developed and trained a model which detects Pneumonia with greater efficacy and efficiency using chest X-Ray images
- ❖ This model is used to detect Pneumonia and detects whether a person is affected with pneumonia or not.
- ❖ This model allows to detect pneumonia with very less cost and greater accuracy and allows to detect Pneumonia in a short period of time
- ❖ This helps to avoid the spreading of one of the deadliest disease with less cost and time.
- ❖ As doctors must do a lot of certain tests to identify if the patient has Pneumonia or not. To solve this cumbersome problem, an ensemble of deep learning models is developed to make the work of doctors easier.
- ❖ This model could help mitigate the reliability and interpretability challenges often faced when dealing with medical imagery.

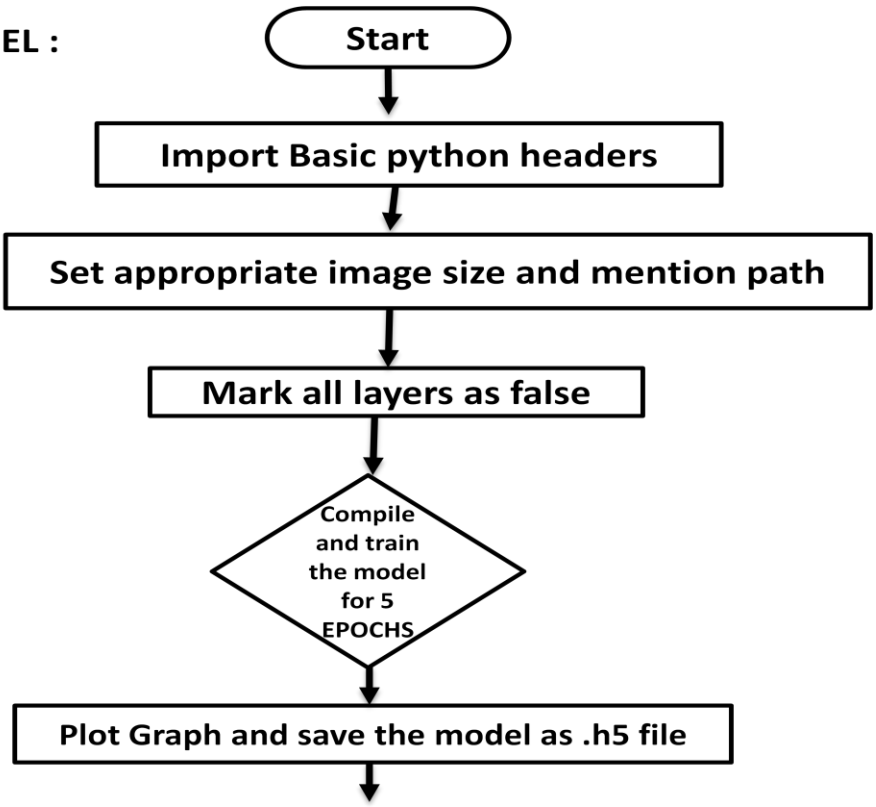
CHAPTER 3

3.1 METHODOLOGY

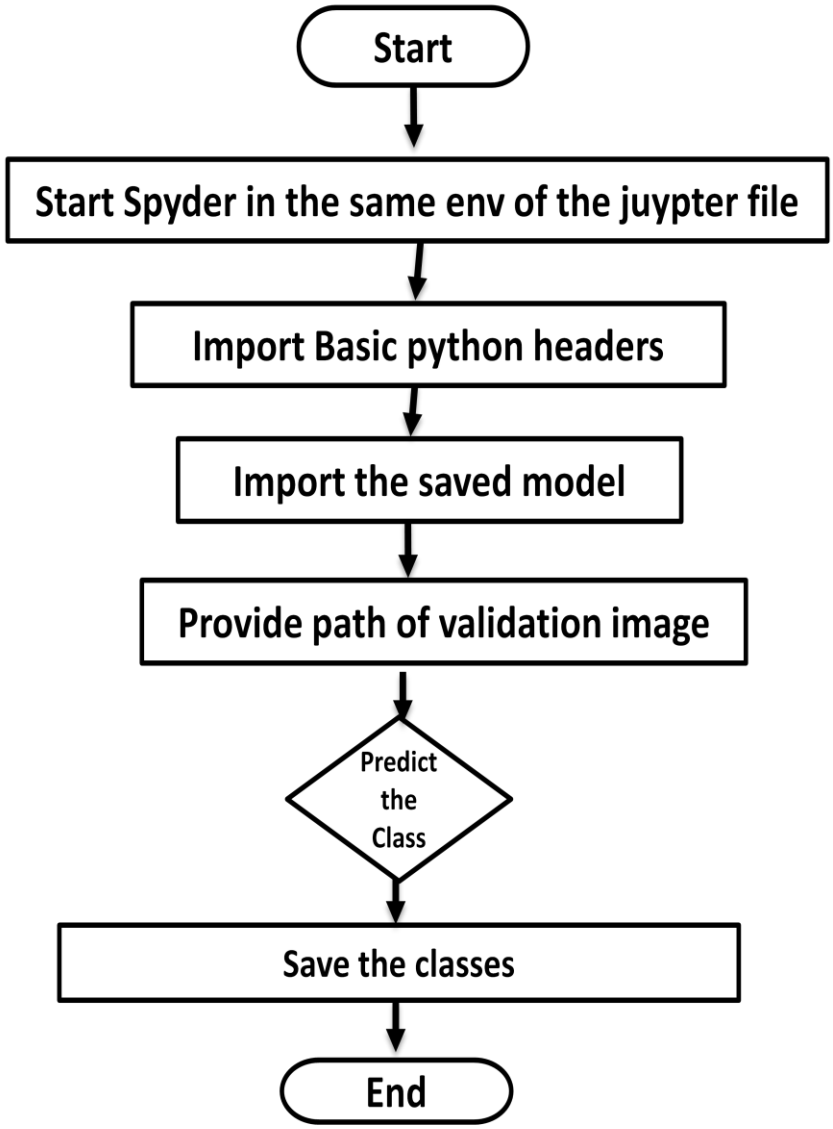
1. Import Essential modules from python pool.
2. Set adequate image size.
3. Make all layers false to train
4. Compile the model
5. Mention exact data set paths accurately
6. Train the model
7. Wait for all epochs to finish
8. Plot graphs
9. Save the model
10. Import model and predict for validation images.

3.2 DESIGN

TRAINING THE MODEL :



PREDICTING USING TRAINED MODEL :



3.3 IMPLEMENTATION (Source code for one of Jupyter file)

```
from keras.layers import Input, Lambda, Dense, Flatten
from keras.models import Model
#from keras.applications.resnet50 import ResNet50
from keras.applications.vgg16 import VGG16
from keras.applications.vgg16 import preprocess_input
from keras.preprocessing import image
from keras.preprocessing.image import ImageDataGenerator
from keras.models import Sequential
import numpy as np
from glob import glob
import matplotlib.pyplot as plt



IMAGE_SIZE = [224, 224]

train_path = 'E:\\chest_xray\\train\\'
valid_path = 'E:\\chest_xray\\test\\'



vgg = VGG16(input_shape=IMAGE_SIZE + [3], weights='imagenet', include_top=False)



for layer in vgg.layers:
    layer.trainable = False



folders = glob('E:\\chest_xray\\train\\*')



prediction = Dense(len(folders), activation='softmax')(x)
model = Model(inputs=vgg.input, outputs=prediction)



model.summary()
```

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

```
from keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(rescale = 1./255,
                                   shear_range = 0.2,
                                   zoom_range = 0.2,
                                   horizontal_flip = True)

test_datagen = ImageDataGenerator(rescale = 1./255)
```

```
training_set = train_datagen.flow_from_directory('E:\\chest_xray\\train',
                                                 target_size = (224, 224),
                                                 batch_size = 32,
                                                 class_mode = 'categorical')


test_set = test_datagen.flow_from_directory('E:\\chest_xray\\test',
                                            target_size = (224, 224),
                                            batch_size = 32,
                                            class_mode = 'categorical')
```

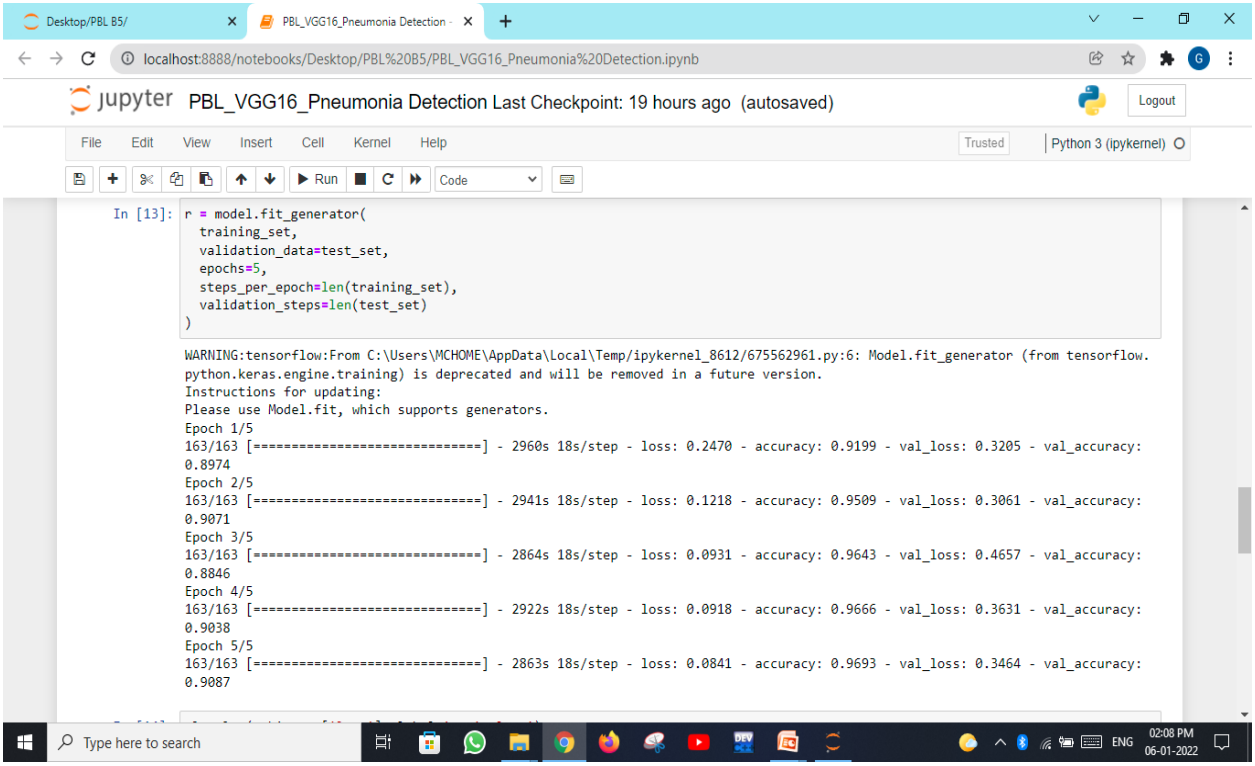
```
r = model.fit_generator(
  training_set,
  validation_data=test_set,
  epochs=5,
  steps_per_epoch=len(training_set),
  validation_steps=len(test_set)
)
```

```
import tensorflow as tf

from keras.models import load_model

model.save('E:\\chest_xray\\val\\model_vgg16.h5')
```

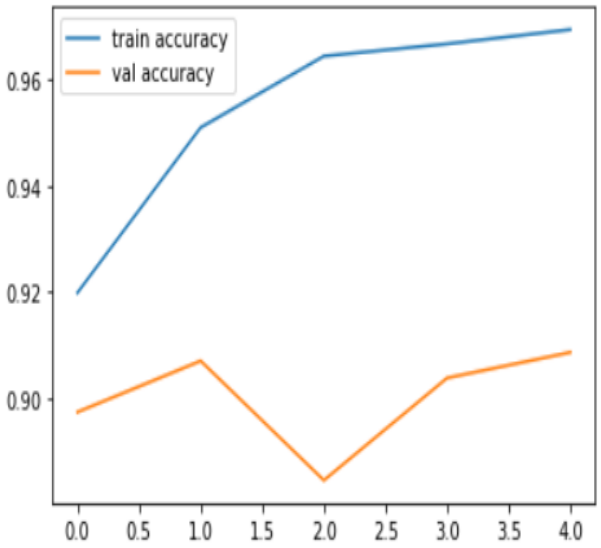
(Source code for one of Spyder file)

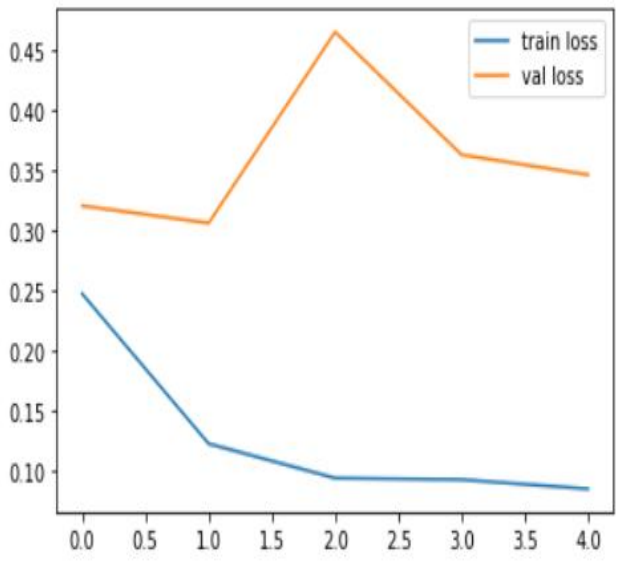
```
from keras.models import load_model

from keras.preprocessing import image

from keras.applications.vgg16 import preprocess_input

import numpy as np


# (1)PREDICTION USING VGG16


model = load_model('C:\\Users\\MCHOME\\Desktop\\PBL B5\\PBL_VGG16\\model_vgg16.h5')


img=image.load_img('E:\\chest_xray\\val\\NORMAL\\NORMAL2-IM-1436-0001.jpeg',target_size=(224,224))


x=image.img_to_array(img)

x=np.expand_dims(x,axis=0)

img_data=preprocess_input(x)

classes=model.predict(img_data)



# (2)PREDICTION USING VGG19

model = load_model('C:\\Users\\MCHOME\\Desktop\\PBL B5\\PBL_VGG19\\model_vgg19.h5')

img=image.load_img('E:\\chest_xray\\val\\NORMAL\\NORMAL2-IM-1436-0001.jpeg',target_size=(224,224))

x=image.img_to_array(img)

x=np.expand_dims(x,axis=0)

img_data=preprocess_input(x)

classes=model.predict(img_data)
```

3.4 RESULTS



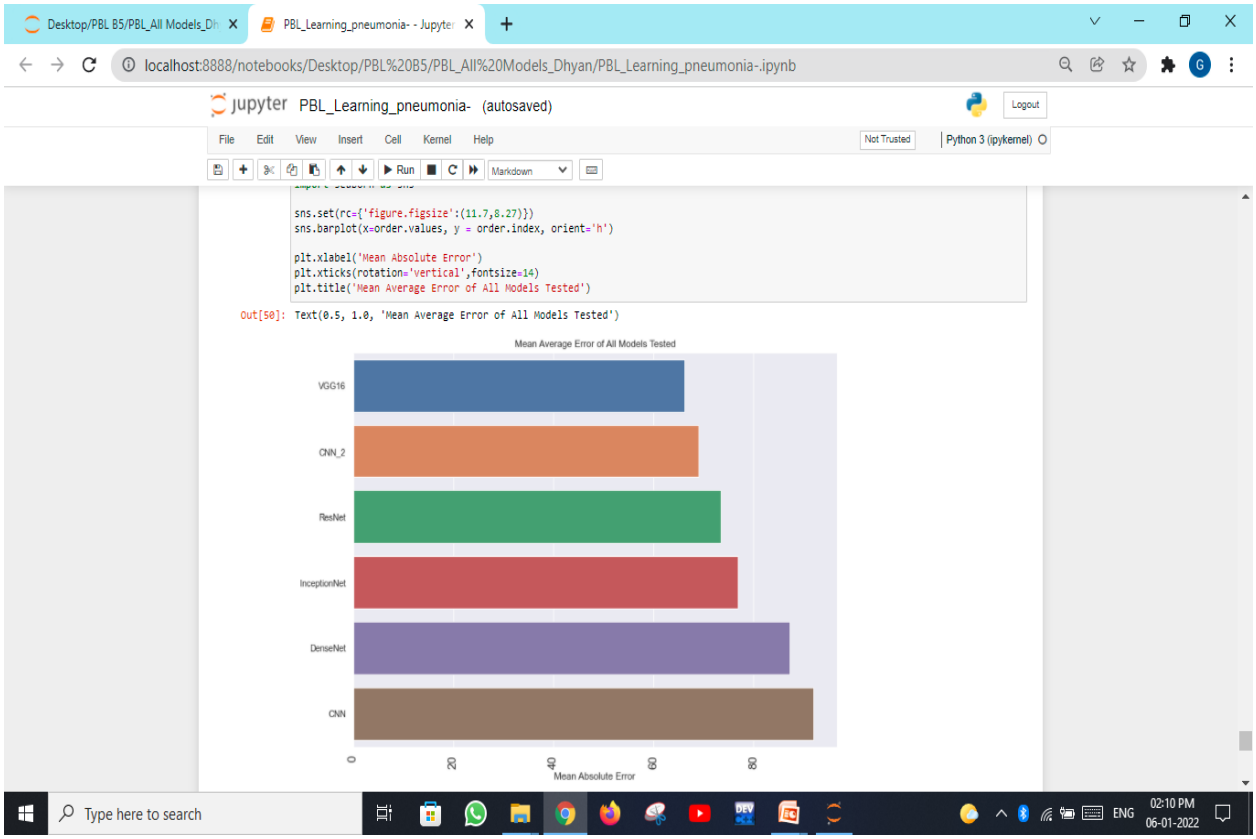

```
# plot the accuracy
plt.plot(r.history['accuracy'], label='train accuracy')
plt.plot(r.history['val_accuracy'], label='val accuracy')
plt.legend()
plt.show()
plt.savefig('AccVal_accuracy')
```

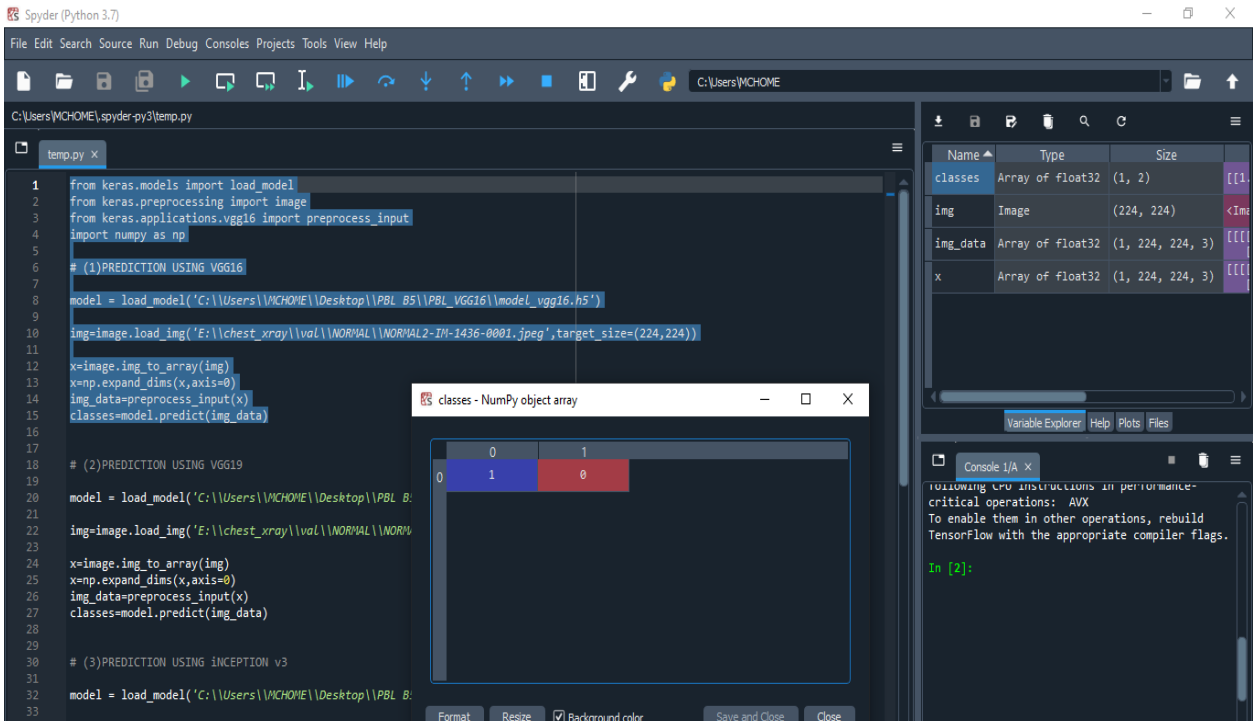


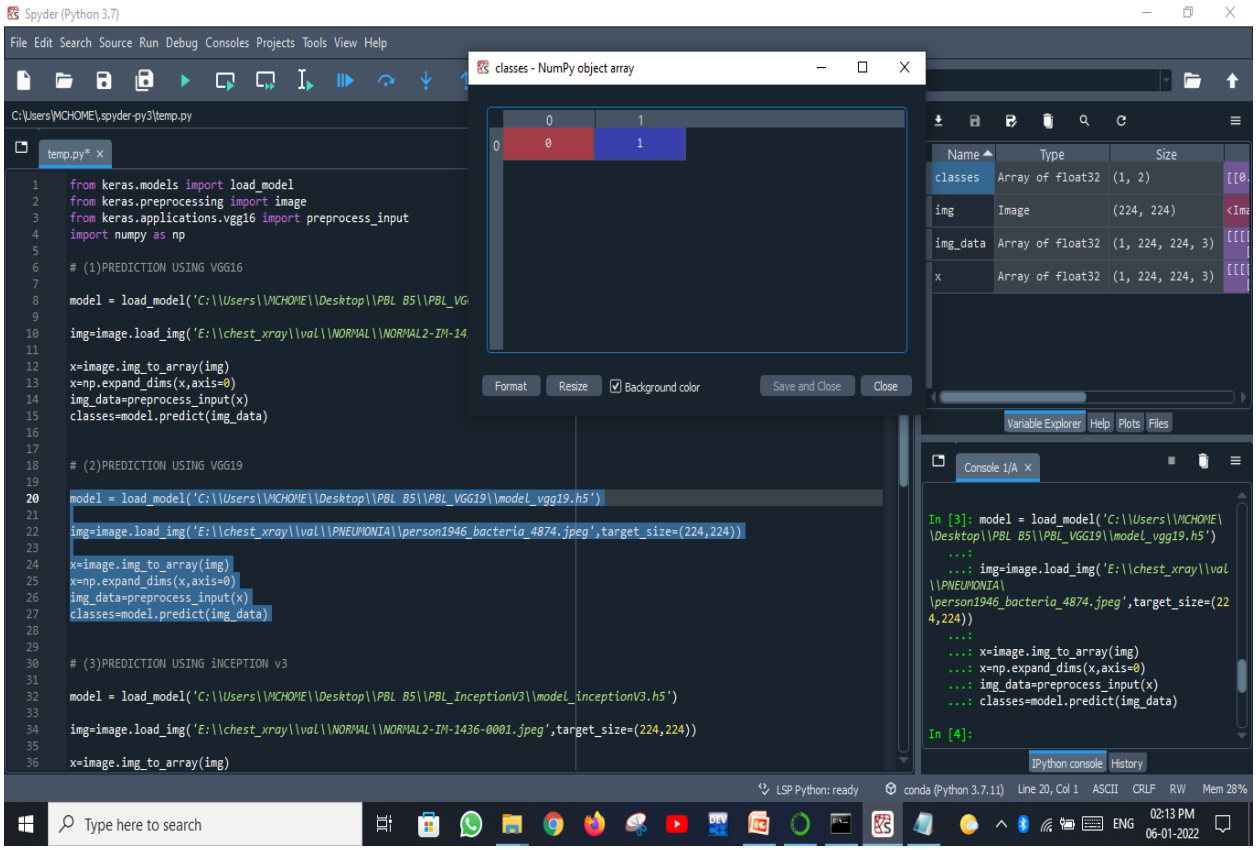


<Figure size 432x288 with 0 Axes>

```
import tensorflow as tf

from keras.models import load_model

model.save('E:\\chest_xray\\val\\model_vgg16.h5')
```


Desktop/PBL B5/PBL_All Models_Dh
PBL_Learning_pneumonia- - Jupyter
localhost:8888/notebooks/Desktop/PBL%20B5/PBL_All%20Models_Dhyan/PBL_Learning_pneumonia-.ipynb
jupyter PBL_Learning_pneumonia- (autosaved)
Logout
File Edit View Insert Cell Kernel Help
Not Trusted
Python 3 (ipykernel)
Run
Markdown
sns.set(rc={'figure.figsize':(11.7,8.27)})
sns.barplot(x=order.values, y = order.index, orient='h')
plt.xlabel('Mean Absolute Error')
plt.xticks(rotation='vertical',fontsize=14)
plt.title('Mean Average Error of All Models Tested')
Out[50]: Text(0.5, 1.0, 'Mean Average Error of All Models Tested')
Mean Average Error of All Models Tested
VGG16
CNN_2
ResNet
InceptionNet
DenseNet
CNN
Mean Absolute Error
Type here to search
ENG
02:10 PM
06-01-2022


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\MCHOME
C:\Users\MCHOME\.spyder-py3\temp.py
temp.py
from keras.models import load_model
from keras.preprocessing import image
from keras.applications.vgg16 import preprocess_input
import numpy as np
# (1)PREDICTION USING VGG16
model = load_model('C:\\Users\\MCHOME\\Desktop\\PBL B5\\PBL_VGG16\\model_vgg16.h5')
img=image.load_img('E:\\chest_xray\\val\\NORMAL\\NORMAL2-IM-1436-0001.jpeg',target_size=(224,224))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
img_data=preprocess_input(x)
classes=model.predict(img_data)
# (2)PREDICTION USING VGG19
model = load_model('C:\\Users\\MCHOME\\Desktop\\PBL B5
img=image.load_img('E:\\chest_xray\\val\\NORMAL\\NORM
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
img_data=preprocess_input(x)
classes=model.predict(img_data)
# (3)PREDICTION USING INCEPTION v3
model = load_model('C:\\Users\\MCHOME\\Desktop\\PBL B5
classes - NumPy object array
0 1
0 1 0
Format Resize Background color Save and Close Close
Name Type Size
classes Array of float32 (1, 2)
img Image (224, 224)
img_data Array of float32 (1, 224, 224, 3)
x Array of float32 (1, 224, 224, 3)
Variable Explorer Help Plots Files
Console 1/A
critical operations: AVX
To enable them in other operations, rebuild
Tensorflow with the appropriate compiler flags.
In [2]:


Spyder (Python 3.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\MCHOME\.spyder-py3\temp.py
temp.py*
from keras.models import load_model
from keras.preprocessing import image
from keras.applications.vgg16 import preprocess_input
import numpy as np
# (1)PREDICTION USING VGG16
model = load_model('C:\\Users\\MCHOME\\Desktop\\PBL B5\\PBL_VG
img=image.load_img('E:\\chest_xray\\val\\NORMAL\\NORMAL2-IM-14
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
img_data=preprocess_input(x)
classes=model.predict(img_data)
# (2)PREDICTION USING VGG19
model = load_model('C:\\Users\\MCHOME\\Desktop\\PBL B5\\PBL_VGG19\\model_vgg19.h5')
img=image.load_img('E:\\chest_xray\\val\\PNEUMONIA\\person1946_bacteria_4874.jpeg',target_size=(224,224))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
img_data=preprocess_input(x)
classes=model.predict(img_data)
# (3)PREDICTION USING iNCEPTION v3
model = load_model('C:\\Users\\MCHOME\\Desktop\\PBL B5\\PBL_InceptionV3\\model_inceptionV3.h5')
img=image.load_img('E:\\chest_xray\\val\\NORMAL\\NORMAL2-IM-1436-0001.jpeg',target_size=(224,224))
x=image.img_to_array(img)
classes - NumPy object array
Format
Resize
Background color
Save and Close
Close
Name
Type
Size
classes
Array of float32
(1, 2)
img
Image
(224, 224)
img_data
Array of float32
(1, 224, 224, 3)
x
Array of float32
(1, 224, 224, 3)
Variable Explorer
Help
Plots
Files
Console 1/A
In [3]: model = load_model('C:\\Users\\MCHOME\\Desktop\\PBL B5\\PBL_VGG19\\model_vgg19.h5')
In [4]:
IPython console
History
LSP Python: ready
conda (Python 3.7.11)
Line 20, Col 1
ASCII
CRLF
RW
Mem 28%
Type here to search
ENG
02:13 PM
06-01-2022

CHAPTER 4

4.1 FUTUTRE ENHANCEMENTS

- Hybrid Models of VGG16 and VGG19 as VGGNet with approximately added 9% of accuracy to current models.
- A front end for easy navigation to validation images and prediction using voice.
- Model with more training session for better accuracy will be developed and trained with even. larger dataset.
- Model which could predict different types of pneumonia and its classification
- Implementation in actual Scenarios in practice.

4.2 CONCLUSION

- Covid-19 is an extremely contagious disease caused by SARS-CoV-2. This directly impacts the lungs' air sacs and causes inflammation in some people leading to pneumonia.
- We aim to develop and train a model which can detect normal vs Covid-19, normal vs bacterial pneumonia and normal vs non-covid-19-viral pneumonia.
- We aim to develop a system which has higher efficiency than the existing system and also provide more accurate results with existing and hybrid trained models.

4.3 REFERENCES

[1] Vaishya R, Javaid M, Khan IH, Haleem A. Artificial Intelligence (AI) applications for COVID-19 pandemic. Diabetes & Metabolic Syndrome: Clin Res Rev. 2020

[2] Gao J, Tian Z, Yang X. Breakthrough: Chloroquine phosphate has shown apparent efficacy in treatment of COVID-19 associated pneumonia in clinical studies. Bioscience trends. 2020.

[3] Obaro SK, Madhi SA. Bacterial pneumonia vaccines and childhood pneumonia: are we winning, refining, or redefining? Lancet Infect Dis. 2006;6(3):150–61.