

APPENDIX

```
from google.colab import drive
drive.mount('/content/drive')

!unzip /content/drive/MyDrive/archive.zip_dl=0

import numpy as np
import matplotlib.pyplot as plt
import keras
import pandas

from keras.preprocessing.image import img_to_array
import os

from keras.preprocessing.image import load_img
from keras.preprocessing.image import ImageDataGenerator
from keras.applications.vgg19 import VGG19, preprocess_input, decode_predictions

training_data_generator= ImageDataGenerator(zoom_range=0.5, shear_range=0.3, rescale=1/255,
horizontal_flip=True)

validation_data_generator= ImageDataGenerator(rescale= 1/255)

train = training_data_generator.flow_from_directory(directory="/content/new plant diseases
dataset(augmented)/New Plant Diseases Dataset(Augmented)/train",target_size=(256,256),batch_size=32)

val = validation_data_generator.flow_from_directory(directory="/content/new plant diseases
dataset(augmented)/New Plant Diseases Dataset(Augmented)/valid",target_size=(256,256),batch_size=32)

from keras.layers import Dense, Flatten
from keras.models import Model
from keras.applications.vgg19 import VGG19
import keras

base_model =VGG19(input_shape=(256,256,3),include_top=False)

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

x =Flatten()(base_model.output)
x= Dense(units=38, activation='softmax')(x)

model =Model(base_model.input, x)

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

from keras.callbacks import ModelCheckpoint, EarlyStopping
es =EarlyStopping(monitor='val_accuracy',min_delta=0.01,patience=3,verbose=1)
```

```

mc
=ModelCheckpoint(filepath="best_model.h",monitor='val_accuracy',min_delta=0.01,patience=3,verbose=1,save_best_only=True)

cb=[es,mc]

his =
model.fit_generator(train,steps_per_epoch=16,epochs=50,verbose=1,callbacks=cb,validation_data=val,validation_steps=16)

from keras.callbacks import ModelCheckpoint, EarlyStopping

es =EarlyStopping(monitor='val_accuracy',min_delta=0.01,patience=3,verbose=1)

mc
=ModelCheckpoint(filepath="best_model.h",monitor='val_accuracy',min_delta=0.01,patience=3,verbose=1,save_best_only=True)

cb=[es,mc]

from keras.models import load_model

model=load_model('/content/best_model.h')

acc =model.evaluate_generator(val)[1]

print(acc)

ref=dict(zip(list(train.class_indices.values()),list(train.class_indices.keys()))))

def prediction(path):

    img=load_img(path,target_size=(256,256))

    i=img_to_array(img)

    im=preprocess_input(i)

    img=np.expand_dims(im,axis=0)

    pred =np.argmax(model.predict(img))

    print(pred)

    print(f"The plant diagnosed as{ref[pred]}")

    path="/content/drive/MyDrive/precaution/"+f"{pred}'+".txt"

    f=open(path)

    print(f.read())

```