AttributeError ...

```
!pip install pyttsx3
!pip install playsound
!pip install pyttsx3
!sudo apt-get install espeak
!ldconfig -p | grep libespeak.so.1
```

```
Reading state information... Done
The following additional packages will be installed:
  espeak-data libespeak1 libportaudio2 libsonic0
The following NEW packages will be installed:
  espeak espeak-data libespeak1 libportaudio2 libsonic0
0 upgraded, 5 newly installed, 0 to remove and 45 not upgraded.
Need to get 1,382 kB of archives.
After this operation, 3,178 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libportaudio2 amd64 19.6.0-1
Get:2 http://archive.ubuntu.com/ubuntu jammy/main amd64 libsonic0 amd64 0.2.0-11build1
Get:3 http://archive.ubuntu.com/ubuntu jammy/universe amd64 espeak-data amd64 1.48.15+d
Get:4 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libespeak1 amd64 1.48.15+df
Get:5 http://archive.ubuntu.com/ubuntu jammy/universe amd64 espeak amd64 1.48.15+dfsg-3
Fetched 1,382 kB in 0s (4,329 kB/s)
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend canno
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (This frontend requires a controlling tty.)
debconf: falling back to frontend: Teletype
dpkg-preconfigure: unable to re-open stdin:
Selecting previously unselected package libportaudio2:amd64.
(Reading database ... 131015 files and directories currently installed.)
Preparing to unpack .../libportaudio2_19.6.0-1.1_amd64.deb ...
Unpacking libportaudio2:amd64 (19.6.0-1.1) ...
Selecting previously unselected package libsonic0:amd64.
Preparing to unpack .../libsonic0_0.2.0-11build1_amd64.deb ...
Unpacking libsonic0:amd64 (0.2.0-11build1) ...
Selecting previously unselected package espeak-data:amd64.
Preparing to unpack .../espeak-data_1.48.15+dfsg-3_amd64.deb ...
Unpacking espeak-data:amd64 (1.48.15+dfsg-3) ...
Selecting previously unselected package libespeak1:amd64.
Preparing to unpack .../libespeak1_1.48.15+dfsg-3_amd64.deb ...
Unpacking libespeak1:amd64 (1.48.15+dfsg-3) ...
Selecting previously unselected package espeak.
Preparing to unpack .../espeak_1.48.15+dfsg-3_amd64.deb ...
Unpacking espeak (1.48.15+dfsg-3) ...
Setting up libportaudio2:amd64 (19.6.0-1.1) ...
Setting up libsonic0:amd64 (0.2.0-11build1) ...
Setting up espeak-data:amd64 (1.48.15+dfsg-3) ...
Setting up libespeak1:amd64 (1.48.15+dfsg-3) ...
Setting up espeak (1.48.15+dfsg-3) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.4) ...
/sbin/ldconfig.real: /usr/local/lib/libtbb.so.12 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbmalloc_proxy.so.2 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbmalloc.so.2 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind_2_5.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind_2_0.so.3 is not a symbolic link

/sbin/ldconfig.real: /usr/local/lib/libtbbbind.so.3 is not a symbolic link

        libespeak.so.1 (libc6,x86-64) => /lib/x86_64-linux-gnu/libespeak.so.1
```

```
!mkdir '/content/Train'
```

```
!mv '/content/Train.zip' '/content/Train/'
```

```
mv: cannot stat '/content/Train.zip': No such file or directory
```

```
!mkdir '/content/Test'
```

```
!mkdir '/content/model'
```

```
!mv '/content/Test.zip' '/content/Test/'
```

```
!unzip '/content/Train/Train.zip'
```

```
Archive:  /content/Train/Train.zip
  inflating: 1Hundrednote/1.jpg
  inflating: 1Hundrednote/10.jpg
  inflating: 1Hundrednote/11.jpg
  inflating: 1Hundrednote/13.jpg
  inflating: 1Hundrednote/18.jpg
  inflating: 1Hundrednote/2.jpg
  inflating: 1Hundrednote/20.jpg
  inflating: 1Hundrednote/21.jpg
  inflating: 1Hundrednote/22.jpg
  inflating: 1Hundrednote/23.jpg
  inflating: 1Hundrednote/24.jpg
  inflating: 1Hundrednote/25.jpg
  inflating: 1Hundrednote/27.jpg
  inflating: 1Hundrednote/28.jpg
  inflating: 1Hundrednote/29.jpg
  inflating: 1Hundrednote/3.jpg
  inflating: 1Hundrednote/30.jpg
  inflating: 1Hundrednote/4.jpg
  inflating: 1Hundrednote/5.jpg
  inflating: 1Hundrednote/6.jpg
  inflating: 1Hundrednote/7.jpg
  inflating: 1Hundrednote/8.jpg
  inflating: 2Hundrednote/1.jpg
  inflating: 2Hundrednote/10.jpg
  inflating: 2Hundrednote/11.jpg
  inflating: 2Hundrednote/12.jpg
  inflating: 2Hundrednote/13.jpg
  inflating: 2Hundrednote/17.jpg
  inflating: 2Hundrednote/18.jpg
  inflating: 2Hundrednote/19.jpg
  inflating: 2Hundrednote/2.jpg
  inflating: 2Hundrednote/20.jpg
  inflating: 2Hundrednote/21.jpg
  inflating: 2Hundrednote/25.jpg
  inflating: 2Hundrednote/26.jpg
  inflating: 2Hundrednote/27.jpg
  inflating: 2Hundrednote/28.jpg
  inflating: 2Hundrednote/29.jpg
  inflating: 2Hundrednote/30.jpg
  inflating: 2Hundrednote/4.jpg
  inflating: 2Hundrednote/5.jpg
  inflating: 2Hundrednote/6.jpg
  inflating: 2Hundrednote/7.jpg
  inflating: 2Hundrednote/8.jpg
  inflating: 2Thousandnote/1.jpg
  inflating: 2Thousandnote/10.jpg
  inflating: 2Thousandnote/11.jpg
  inflating: 2Thousandnote/12.jpg
  inflating: 2Thousandnote/13.jpg
  inflating: 2Thousandnote/16.jpg
  inflating: 2Thousandnote/17.jpg
  inflating: 2Thousandnote/19.jpg
  inflating: 2Thousandnote/2.jpg
  inflating: 2Thousandnote/20.jpg
  inflating: 2Thousandnote/21.jpg
  inflating: 2Thousandnote/22.jpg
  inflating: 2Thousandnote/24.jpg
```

```
!unzip '/content/Test/Test.zip'
```

You

Please explain the error:

```
AttributeError:
```

Colab AI

Hypothesis:

The error is caused by the `plot_model` function. The function expects a single Keras model as its first argument, but th user is passing a list of models (`models`). This is not supported by the function, hence the error message.

Suggested Changes:

To fix the error, the user should pass each mode to the `plot_model` function individually. Here's the corrected code:

```
from tensorflow

# Plot each mod
for i, model in
    plot_model(
```

Use code with caution

This code iterates through the list of models and plots each model to a

```
Archive:  /content/Test/Test.zip
   creating: Test/1Hundrednote/
  inflating: Test/1Hundrednote/1.jpg
  inflating: Test/1Hundrednote/14.jpg
  inflating: Test/1Hundrednote/15.jpg
  inflating: Test/1Hundrednote/16.jpg
  inflating: Test/1Hundrednote/2.jpg
  inflating: Test/1Hundrednote/3.jpg
   creating: Test/2Hundrednote/
  inflating: Test/2Hundrednote/1.jpg
  inflating: Test/2Hundrednote/2.jpg
  inflating: Test/2Hundrednote/3.jpg
  inflating: Test/2Hundrednote/31.jpg
  inflating: Test/2Hundrednote/32.jpg
  inflating: Test/2Hundrednote/33.jpg
   creating: Test/2Thousandnote/
  inflating: Test/2Thousandnote/1.jpg
  inflating: Test/2Thousandnote/2.jpg
  inflating: Test/2Thousandnote/3.jpg
  inflating: Test/2Thousandnote/31.jpg
  inflating: Test/2Thousandnote/32.jpg
  inflating: Test/2Thousandnote/33.jpg
   creating: Test/5Hundrednote/
  inflating: Test/5Hundrednote/1.jpg
  inflating: Test/5Hundrednote/2.jpg
  inflating: Test/5Hundrednote/3.jpg
  inflating: Test/5Hundrednote/31.jpg
  inflating: Test/5Hundrednote/32.jpg
  inflating: Test/5Hundrednote/33.jpg
   creating: Test/Fiftynote/
  inflating: Test/Fiftynote/1.jpg
  inflating: Test/Fiftynote/2.jpg
  inflating: Test/Fiftynote/27.jpg
  inflating: Test/Fiftynote/28.jpg
  inflating: Test/Fiftynote/29.jpg
  inflating: Test/Fiftynote/3.jpg
   creating: Test/Tennote/
  inflating: Test/Tennote/1.jpg
  inflating: Test/Tennote/2.jpg
  inflating: Test/Tennote/3.jpg
  inflating: Test/Tennote/31.jpg
  inflating: Test/Tennote/32.jpg
  inflating: Test/Tennote/33.jpg
   creating: Test/Twentynote/
  inflating: Test/Twentynote/1.jpg
  inflating: Test/Twentynote/18.jpg
  inflating: Test/Twentynote/2.jpg
  inflating: Test/Twentynote/24.jpg
  inflating: Test/Twentynote/3.jpg
  inflating: Test/Twentynote/30.jpg
```

```
import os
import numpy as np
import matplotlib.pyplot as plt
import random
import cv2
import PIL
import glob
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.preprocessing.image import ImageDataGenerator


ROOTPATH = '/content'
DATAPATH= ROOTPATH+'/Train'
TRAINPATH = ROOTPATH+'/Train'
TESTPATH =  ROOTPATH+'/Test'
MODELPATH = ROOTPATH+'/model/'
```

separate image file. The `f-string` is used to generate unique file names for each plot.

Rate this answer

```
_1Hundrednote=glob.glob(DATAPATH+'/1Hundrednote/*')# [/content/Test/1Hundrednote/1.jpg,/conten
_2Hundrednote=glob.glob(DATAPATH+'/2Hundrednote/*')
_2Thousandnote=glob.glob(DATAPATH+'/2Thousandnote/*')
_5Hundrednote=glob.glob(DATAPATH+'/5Hundrednote/*')
_Fiftynote=glob.glob(DATAPATH+'/Fiftynote/*')
_Tennote=glob.glob(DATAPATH+'/Tennote/*')
_Twentynote=glob.glob(DATAPATH+'/Twentynote/*')


print(len(_1Hundrednote),_1Hundrednote)
print(len(_2Hundrednote),_2Hundrednote)
print(len(_2Thousandnote),_2Thousandnote)
print(len(_2Thousandnote),_2Thousandnote)
print(len(_5Hundrednote),_5Hundrednote)
print(len(_Fiftynote),_Fiftynote)
print(len(_Tennote),_Tennote)
print(len(_Twentynote),_Twentynote)
```

```
22 ['/content/Train/1Hundrednote/10.jpg', '/content/Train/1Hundrednote/27.jpg', '/content/
22 ['/content/Train/2Hundrednote/10.jpg', '/content/Train/2Hundrednote/27.jpg', '/content/
21 ['/content/Train/2Thousandnote/10.jpg', '/content/Train/2Thousandnote/27.jpg', '/conten
21 ['/content/Train/2Thousandnote/10.jpg', '/content/Train/2Thousandnote/27.jpg', '/conten
22 ['/content/Train/5Hundrednote/9.jpg', '/content/Train/5Hundrednote/19.jpg', '/content/T
22 ['/content/Train/Fiftynote/19.jpg', '/content/Train/Fiftynote/6.jpg', '/content/Train/F
22 ['/content/Train/Tennote/10.jpg', '/content/Train/Tennote/27.jpg', '/content/Train/Tenn
22 ['/content/Train/Twentynote/10.jpg', '/content/Train/Twentynote/9.jpg', '/content/Train
```

```
dataset_classes=[_1Hundrednote,_2Hundrednote,_2Thousandnote,_5Hundrednote,_Fiftynote,_Tennote,
total_class=len(dataset_classes)
print('Total dataset class: ',total_class)
```

```
Total dataset class:  7
```

```
IMAGE_SIZE=224
BATCH_SIZE=64

#pre_processing_training
train_datagen = ImageDataGenerator(
        rescale=1./255,
        shear_range=0.2,
        zoom_range=0.2,
        horizontal_flip=True,
        fill_mode='nearest',
        validation_split=0.2)

training_set = train_datagen.flow_from_directory(
        DATAPATH,
        shuffle=True,
        target_size=(IMAGE_SIZE,IMAGE_SIZE),
        batch_size=BATCH_SIZE,
        class_mode='categorical',
        subset='training')


validation_set = train_datagen.flow_from_directory(
        DATAPATH,
         shuffle=True,
        target_size=(IMAGE_SIZE,IMAGE_SIZE),
        batch_size=BATCH_SIZE,
        class_mode='categorical',
        subset='validation')
```

```
Found 125 images belonging to 8 classes.
Found 28 images belonging to 8 classes.
```

```
test_datagen = ImageDataGenerator(rescale=1./255)
test_set = test_datagen.flow_from_directory(
        TESTPATH,
         shuffle=False,
        target_size=(IMAGE_SIZE,IMAGE_SIZE),
        batch_size=BATCH_SIZE,
        class_mode='categorical')
```

```
Found 42 images belonging to 8 classes.
```

```
training_set.class_indices
```

```
{'.ipynb_checkpoints': 0,
 '1Hundrednote': 1,
 '2Hundrednote': 2,
 '2Thousandnote': 3,
 '5Hundrednote': 4,
 'Fiftynote': 5,
 'Tennote': 6,
 'Twentynote': 7}
```

```
validation_set.class_indices
```

```
{'.ipynb_checkpoints': 0,
 '1Hundrednote': 1,
 '2Hundrednote': 2,
 '2Thousandnote': 3,
 '5Hundrednote': 4,
 'Fiftynote': 5,
 'Tennote': 6,
 'Twentynote': 7}
```

```
test_set.class_indices
```

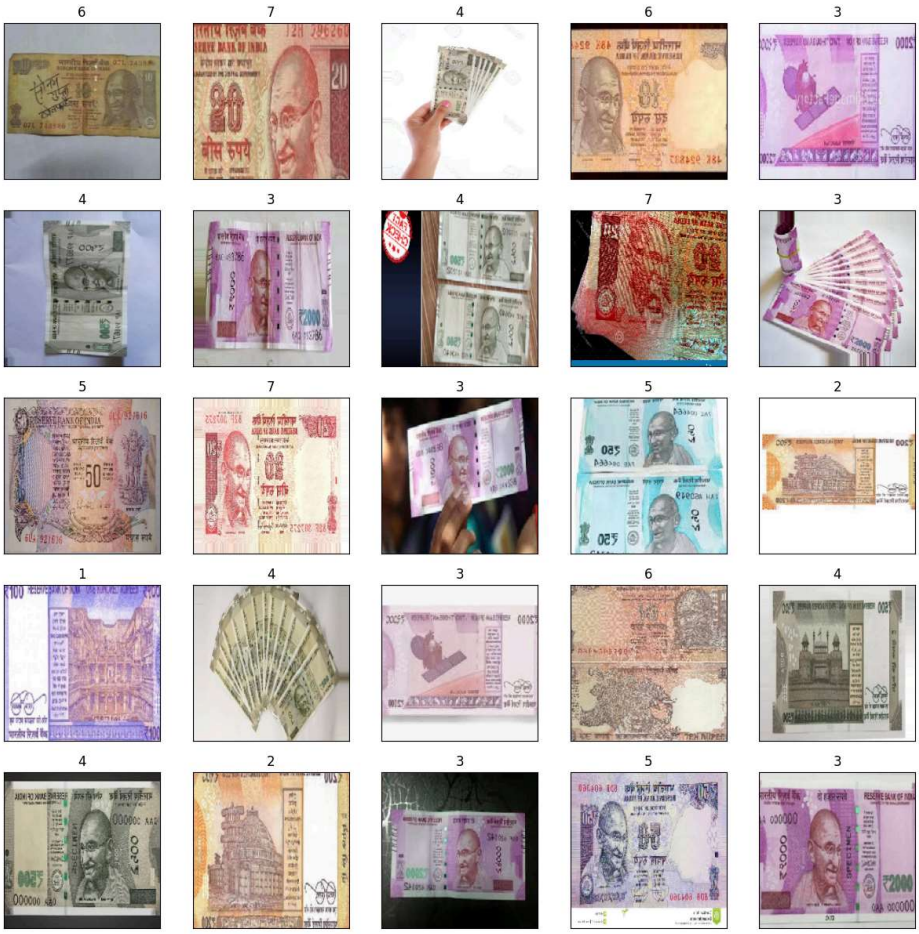
```
{'.ipynb_checkpoints': 0,
 '1Hundrednote': 1,
 '2Hundrednote': 2,
 '2Thousandnote': 3,
 '5Hundrednote': 4,
 'Fiftynote': 5,
 'Tennote': 6,
 'Twentynote': 7}
```

```
total_class=len(training_set.class_indices)
print('Number of classes in dataset: ',total_class)
```

```
Number of classes in dataset:  8
```

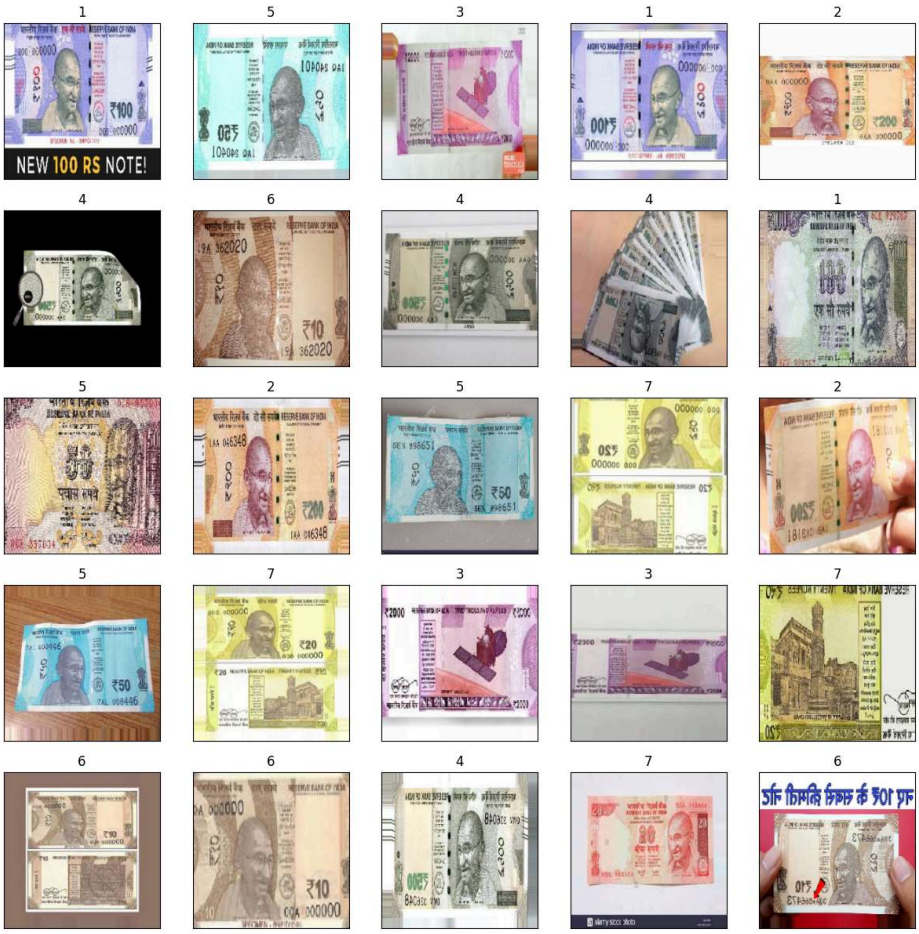
```
x,y=training_set.next()
fig=plt.figure(figsize=(15,15))
rows=5
cols=5
for i in range(rows*cols):
    fig.add_subplot(rows,cols,i+1)
    image=x[i]
    plt.imshow(image)
    plt.title(np.argmax(y[i]))
    plt.xticks([])
    plt.yticks([])

plt.show()
```




```
x,y=validation_set.next()
fig=plt.figure(figsize=(15,15))
rows=5
cols=5
for i in range(rows*cols):
    fig.add_subplot(rows,cols,i+1)
    image=x[i]
    plt.imshow(image)
    plt.title(np.argmax(y[i]))
    plt.xticks([])
    plt.yticks([])

plt.show()
```




```
from tensorflow.keras.models import load_model
from sklearn.metrics import accuracy_score


# !unzip '/content/5.Model_Xception_Pretrained_updated.zip'
```

```
Archive:  /content/5.Model_Xception_Pretrained_updated.zip
  End-of-central-directory signature not found.  Either this file is not
  a zipfile, or it constitutes one disk of a multi-part archive.  In the
  latter case the central directory and zipfile comment will be found on
  the last disk(s) of this archive.
unzip:  cannot find zipfile directory in one of /content/5.Model_Xception_Pretrained_updat
        /content/5.Model_Xception_Pretrained_updated.zip.zip, and cannot find /content/5.M
```

```
model1=load_model('/content/1.CNN1.h5')
model2=load_model('/content/5.Model_Xception_Pretrained_updated.h5')


models=[model1,model2]


for model in models:
    model.summary()
```

```
Model: "sequential_3"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_9 (Conv2D)           (None, 111, 111, 64)      1792

 max_pooling2d_9 (MaxPoolin  (None, 55, 55, 64)        0
 g2D)

 conv2d_10 (Conv2D)          (None, 55, 55, 32)        18464

 max_pooling2d_10 (MaxPooli  (None, 27, 27, 32)        0
 ng2D)

 conv2d_11 (Conv2D)          (None, 27, 27, 32)        9248

 max_pooling2d_11 (MaxPooli  (None, 13, 13, 32)        0
 ng2D)

 flatten_3 (Flatten)         (None, 5408)              0

 dense_6 (Dense)             (None, 128)               692352

 dense_7 (Dense)             (None, 8)                 1032

=================================================================
Total params: 722888 (2.76 MB)
Trainable params: 722888 (2.76 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 xception (Functional)       (None, 2048)              20861480

 flatten (Flatten)           (None, 2048)              0

 dense (Dense)               (None, 8)                 16392

=================================================================
Total params: 20877872 (79.64 MB)
Trainable params: 16392 (64.03 KB)
Non-trainable params: 20861480 (79.58 MB)
_________________________________________________________________
```

```
preds = [model.predict(test_set) for model in models]
preds=np.array(preds)
summed = np.sum(preds, axis=0)

# argmax across classes
ensemble_prediction = np.argmax(summed, axis=1)

prediction1 = np.argmax(model1.predict(test_set),axis=1)
prediction2 = np.argmax(model2.predict(test_set),axis=1)

accuracy1 = accuracy_score(test_set.classes, prediction1)
accuracy2 = accuracy_score(test_set.classes, prediction2)
```

```
from tensorflow.keras.utils import plot_model

# Plot each model and save it to an image file
for i, model in enumerate(models):
    plot_model(model, to_file='ensemble_model_{i+1}.png', show_shapes=True, show_layer_names=Tru
```

```
print('Accuracy Score for average ensemble = ', ensemble_accuracy)
```

```
 1/1 [==============================] - 1s 1s/step
 1/1 [==============================] - 14s 14s/step
 1/1 [==============================] - 1s 853ms/step
 1/1 [==============================] - 11s 11s/step
 Accuracy Score for model1 =  0.8333333333333334
 Accuracy Score for model2 =  0.5476190476190477
```