# EARLYDETECTIONOFFORESTFIREUSINGDEEPLEARNING

## MODEL BUILDING

## ADDINGCNNLAYERS

| | |
|---|---|
| TeamID | PNT2022TMID30386 |
| ProjectName | Project-Earlydetectionofforestfireusingdeep learning |

## ADDINGCNNLAYERS:

WewillbeaddingthreelayersforCNN

- Convolution layer
- Poolinglayer
- Flatteninglayer

**AddingConvolutionalLayer:**

TheconvolutionallayeristhefirstandcorelayerofCNN.Itisoneofthebuildingblocksofa CNN and is used for extracting important features from the image.

IntheConvolutionoperation,theinputimagewillbeconvolvedwiththefeaturedetector/filters to get a feature map. The important role of the feature detector is to extract the features from the image. The group of feature maps is called a feature layer.

In the convolution2D function, we gave arguments that include 32,(3,3), that refers to we are applying 32 filters of 3x3 matrix filter, and input_shape is the input image shape with RGB, here 64x64 is the size and 3 represent the channel, RGB colour images.

Activation Function:These are the functions that help us to decide if we need to activate the nodeornot.Thesefunctionsintroducenon-linearityinthenetworks.

**AddingPoolingLayer**

**MaxPooling**selectsthemaximumelementfromtheregionofthefeaturemapcoveredbythe filter. Thus, the output after max-pooling layer would be a feature map containing the most prominent features of the previous feature map.

After the convolution layer, a pooling layer is added. Max pooling layer can be added using MaxPooling2Dclass.Ittakesthepoolsizeasaparameter.Efficientsizeofthepoolingmatrix is (2,2). It returns the pooled feature maps. (Note:Any number of convolution layers, pooling and dropout layers can be added)

Intheabovecode,pool_sizereferstopoolingfilterorkernelsize.

**Task3:AddingFlattenLayer**

Nowthepooledfeaturemapfromthepoolinglayerwillbeconvertedintoonesingledimension matrix or map, where each pixel in one single column, nothing but flattening. The flattening layer converts the multi-dimension matrix to one single dimension layer.

**IMPORTLIBRARIES:**

11/7/22, 12:35 AM Untitled8.ipynb - Colaboratory

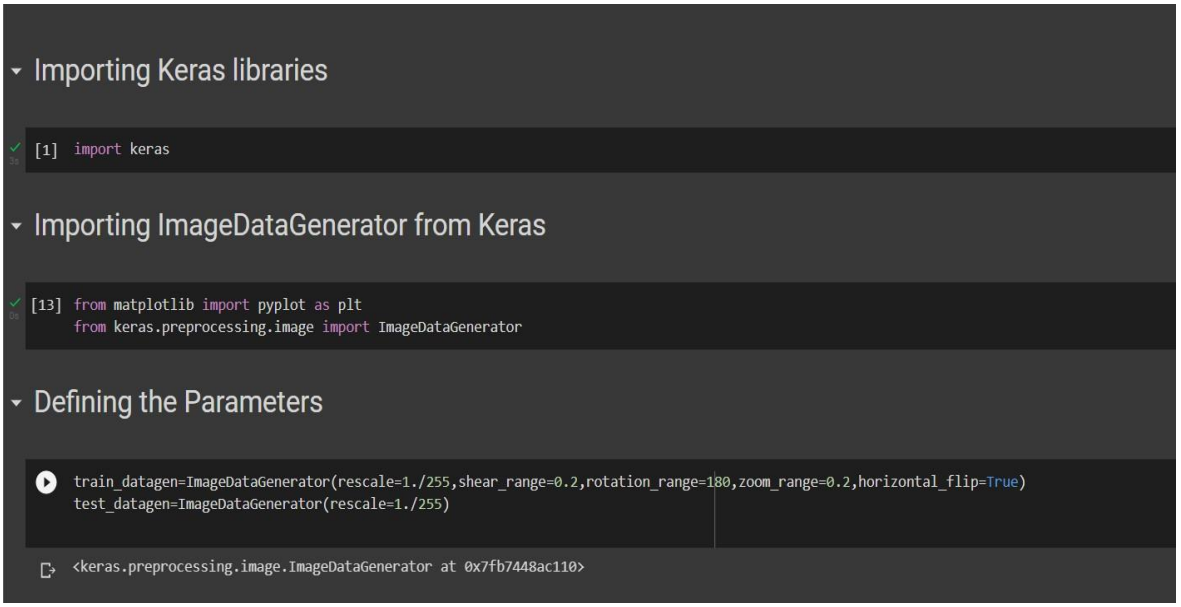
## Importing Keras libraries

```
import keras
```

## Importing ImageDataGenerator from Keras

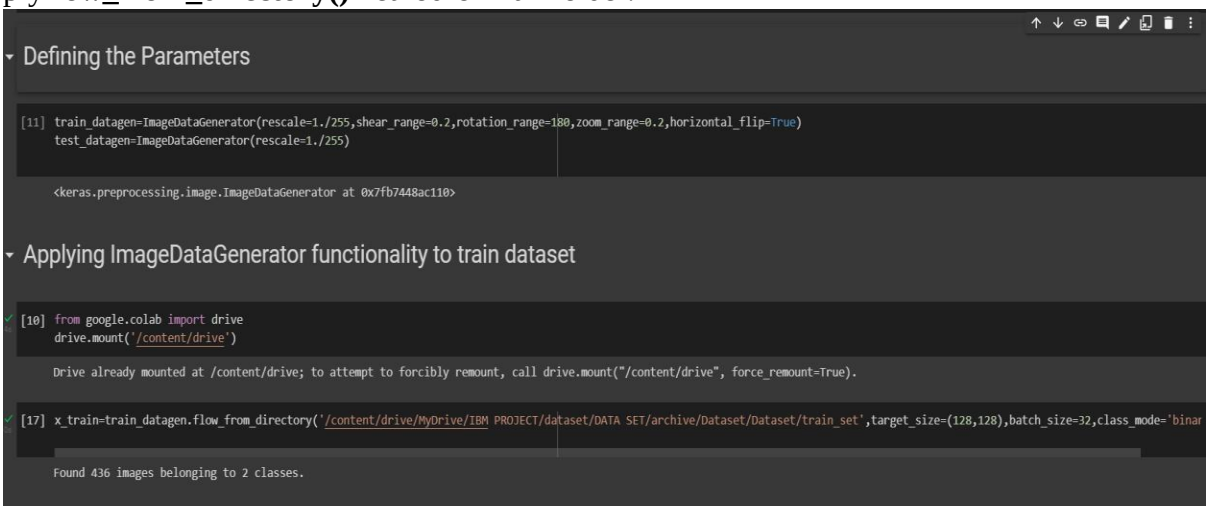
```
from keras.preprocessing.image import ImageDataGenerator
```

**IMPORTImageDataGeneratorFROMKERAS:**



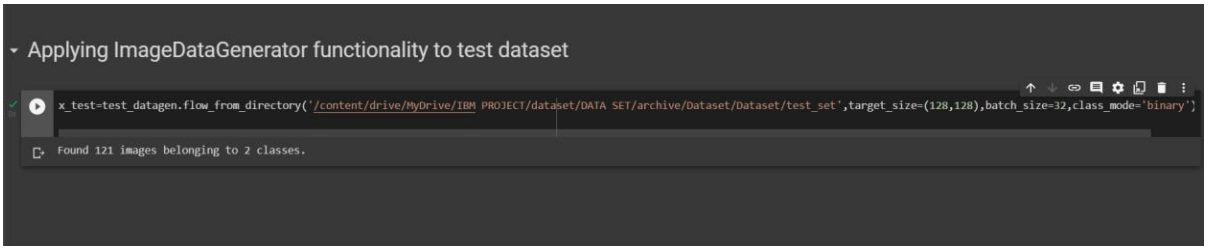
## APPLYINGImageDataGeneratortotraindataset:

ply**flow_from_directory()**methodforTrainfolder.




## APPLYINGImageDataGeneratortotestdataset:

Applyingthe**flow_from_directory ()**methodfortestfolder.




## IMPORTINGMODELBUILDINGLIBRARIES:



Importing Model Building Libraries

```
#to define the linear Initialisation import sequential
from keras.models import Sequential
#to add layers import Dense
from keras.layers import Dense
#to create Convolutional kernel import convolution2D
from keras.layers import Convolution2D
#import Maxpooling layer
from keras.layers import MaxPooling2D
#import flatten layer
from keras.layers import Flatten
import warnings
warnings.filterwarnings('ignore')
```

**INITIALIZINGTHEMODEL:**

## Initializing the model

```
model=Sequential()
```

**ADDINGCNNLAYERS:**

## Adding CNN Layers

```
model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))
#add maxpooling layers
model.add(MaxPooling2D(pool_size=(2,2)))
#add faltten layer
model.add(Flatten())
```