EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRES

MODEL BUILDING

ADDINGCNNLAYERS

TeamID	PNT2022TMID30362
ProjectName	Project- Emerging methods for Early detection of forestfires

ADDINGCNNLAYERS:

WewillbeaddingthreelayersforCNN

- Convolution layer
- Poolinglayer
- Flatteninglayer

AddingConvolutionalLayer:

The convolutional layer is the first and core layer of CNN. It is one of the building blocks of a 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. These functions introduce non-linearity in the networks.

AddingPoolingLayer

MaxPoolingselectsthemaximumelementfromtheregionofthefeaturemapcoveredbythe 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)

 $In the above code, pool_size refers topooling filter or kernel size.\\$

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

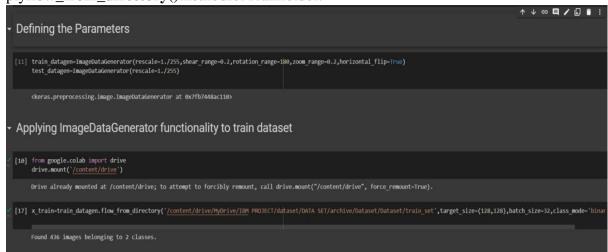
from keras.preprocessing.image import ImageDataGenerator

IMPORTImageDataGeneratorFROMKERAS:

Importing Keras libraries
 Importing ImageDataGenerator from Keras
 Importing ImageDataGenerator from Keras
 In from matplotlib import pyplot as plt from keras.preprocessing.image import ImageDataGenerator
 Defining the Parameters
 train_datagen=ImageDataGenerator(rescale=1./255, shear_range=0.2, rotation_range=180, zoom_range=0.2, horizontal_flip=True) test_datagen=ImageDataGenerator(rescale=1./255)
 keras.preprocessing.image.ImageDataGenerator at 0x7fb7448ac110>

APPLYINGImageDataGeneratortotraindataset:

plyflow_from_directory()methodforTrainfolder.



APPLYINGImageDataGeneratortotestdataset:

Applying the **flow_from_directory** () method for testfolder.



IMPORTINGMODELBUILDINGLIBRARIES:

11/8/22, 1:16 AM

Main code - Colaboratory

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())
```