EMERGING METHODS FOR EARLY DETECTION OF FORESTFIRES

MODEL

BUILDINGADDINGDENS

ELAYERS

TeamID	PNT2022TMID30362
ProjectName	Project-Emerging Method For Early detection of forest fire

ADDINGDENSELAYERS:

The name suggests that layers are fully connected (dense) by the neurons in a network layer. Each neuron in alayer receives input from all the neurons present in the previous layer. Dense is used to add the layers.

AddingHiddenlayers:

This step is to add a dense layer (hidden layer). We flatten the feature map and convert it into a vector or single dimensional array in the Flatten layer. This vector array is fed it as an input to the neural network and applies an activation function, such as sigmoid or other, and returns the output.

Addingoutputlayer:

This step is to add a dense layer (output layer) where you will be specifying the number of classes your dependent variable has, activation function and weight initializer as the arguments. We use add () method to add dense layers. In this layer, no need of mentioning input dimensions as we have mentions them in the above layer itself.

IMPORTLIBRARIES:

11/7/22, 12:35 AM

Untitled8.ipynb - Colaboratory

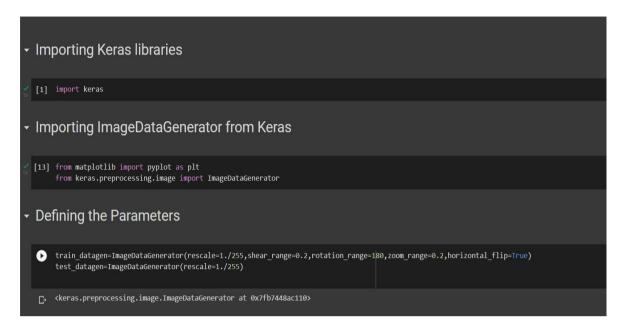
Importing Keras libraries

import keras

Importing ImageDataGenerator from Keras

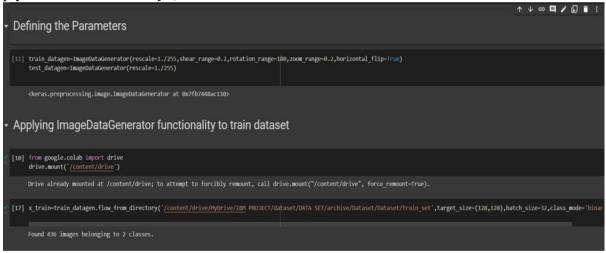
from keras.preprocessing.image import ImageDataGenerator

IMPORTImageDataGenerator FROMKERAS:



APPLYINGImageDataGeneratortotraindataset:

plyflow_from_directory ()methodforTrainfolder.



${\bf APPLYING Image Data Generator to test dataset:}$

Applyingtheflow_from_directory()methodfortestfolder.



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())
```

ADDINGDENSELAYERS:

→ Add Dense layers

```
#add hidden layers
model.add(Dense(150,activation='relu'))
#add output layer
model.add(Dense(1,activation='sigmoid'))
```