**DAY 1**

* **Teachable Machine**

1. Teachable machine is a tool which is used to provide a data set by using class name and we can train the model based on the provided data set it will performs the prediction operation on it will provide the prediction of the output.
2. We have two types of the data class binary and multiple dataclass
3. upload images for both classes
4. train the model(epoch)it means how many times the data is trained
5. UPLOAD DATA---->TRAIN DATA---->TEST
6. TensorFlow Lite(mobile android application)
7. Tensorflow.js(front end backend web development)

* **Machine learning for kids:**

1. In this we will provide the text data in which it will predict the positive negative or neutral output based on the probability score.

**TASK 1**

<https://teachablemachine.withgoogle.com/models/o1xAN_0g3/>

**DIFFERENT TYPES OF LAYERS :**

**Input layer:**

**Convolation Layer:**

**Activatio Layer:**

**Pooling Layer:**

**Output layer:**

**Input layer:**

Provide input such as images,text,etc.

We will provide the size of the image include width hight etc

**Convolation Layer:**

An convolution layer we apply Filters to get featuers(Kernel).

Filter is used to recognize the image properly

Shape of the image is called feature

We will map the feature

**Activatio Layer:**

To activate the inactive neurons an activation function is used to activate the neurons,

An activation function is present in this layer.

Activation Function

-Tanh

-sigmoid

-relu

-softmax

**Pooling Layer:**

It is periodicallay inserted wherever needed

Main function is to reduce the size of volume which makes the computation fast reduces memory and also prevents overfitting.

**Practical:**

Select 2\*2 block take the maximum value for all the three feature

**Types:**

-Max pooling

-average pooling

Training time depends on pooling layer

**Output layer:**

The output layer from the fully connected layers is the fed into a logistic function for classification tasks like sigmoid or softmax which converts the output of each class into the probability score{ex: When we give jai ballaya it give negative with 100% confidence ,here 100%confidence(Accuressy) is probability score} of each class

-if the input >0.5 then it will give positive(1)

<0.5 then it will give negative(0)

More then ten(10) class the result obtained is Accurate.

We save the file with .h5 file because it is trained model, after trained data it will be saved as module.h5 extension file for **feature purpose**

**LEARN THIS TODAY :**

TENSORFLOW

KERAS

IMAGEDATAGENERATOR

Convert the images in the form of grey scale image(Validation=Correct catogary or not)

Link : It is used to provide the input images from Drive.(.flow\_from\_directory)

Target size = image size

Batch size = Set of images

Class mode : Binary

Subset = ‘Validation’

Code to mount to Gdrive

from google.colab import drive

drive.mount('/content/drive')

**ASSAINMENT FOR TODAY..........(24/06/24)**

20 image for class1 and class2 (train test the data model google image resizer

**DAY 2**

**Started with practical of the Convolational layer and all the layer**

**—-------------------------------------------25/06/2024—---------------------------------**

**TRICKIER CASES**

**translation**

**.scaling**

**.rotation**

**Weight**

**POOLING(SELECT THE MAXIMUM VALUE OF A EVERY FEATURE)**

**RELU(Rectified Linear Units):**

It will convert the negative values to zero

The output of one becomes the input of the another.

**Deep Stacking :**

Layers can be repeated several no of times

Fully connected layer ;

Flatten layer ; 2d to 1d array

Based on the index no which have the value 1 predict X or 0

Keras.sequential,sequential is used because it is connected with many layers

Algorithm is ready we have to compile (model.compile)

Parameters : optimizer, loss,metrics

Optimiser : during the compilation it will remove unwanted filters unwanted things and it will perform conversion of high level language to machine level language based on the loss and accuracy

Mode.save : save the model with.h5 it will automettically save in file

Download and upload in the google drive because if we want to do extra operation we cannot run from first we can from wherewe stopped

Load\_models and image(keras , PIL) are sub packages imported for output side

**STEPS :**

Load the images(link).

Give the image path(link).

Image to Array ,Array ton image we use numpy

Finally with help of the matrix only we predict the output..

**CONVOLUTION—---------POOLING—---------RELUs**

**FULLY CONNECTED LAYER—-DENSE LAYER**

**OUTPUT OF CONVOLUTION LAYER ALWAYS WILL BE 1-DIMENSIONAL ARRAY it is also called as flatten layer**

**—---------------------------------------------------------------**

**Optimiser—it acts as a compiler and removes unwanted things**

**OUTPUT FUNCTION**

**Import the load model, image and numpy**

MIT App Invevtor(mobile app)