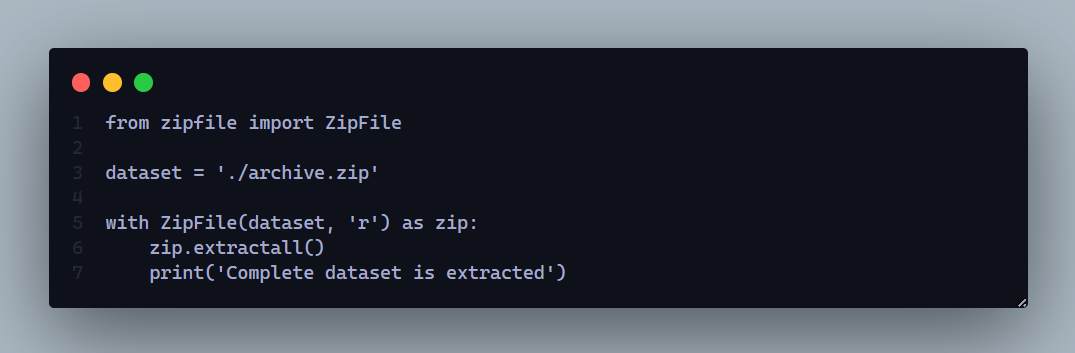
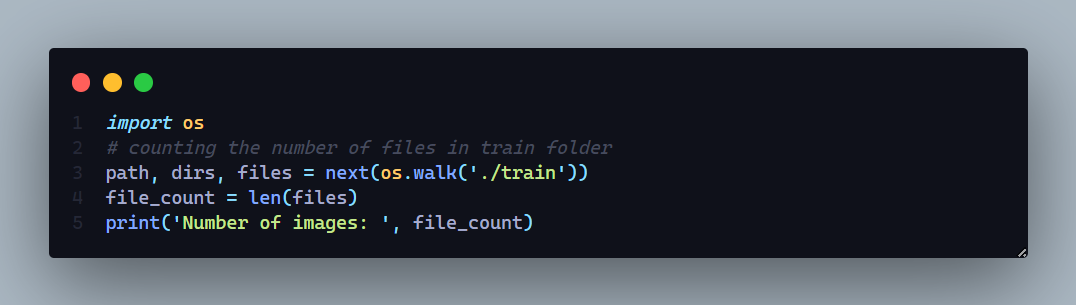
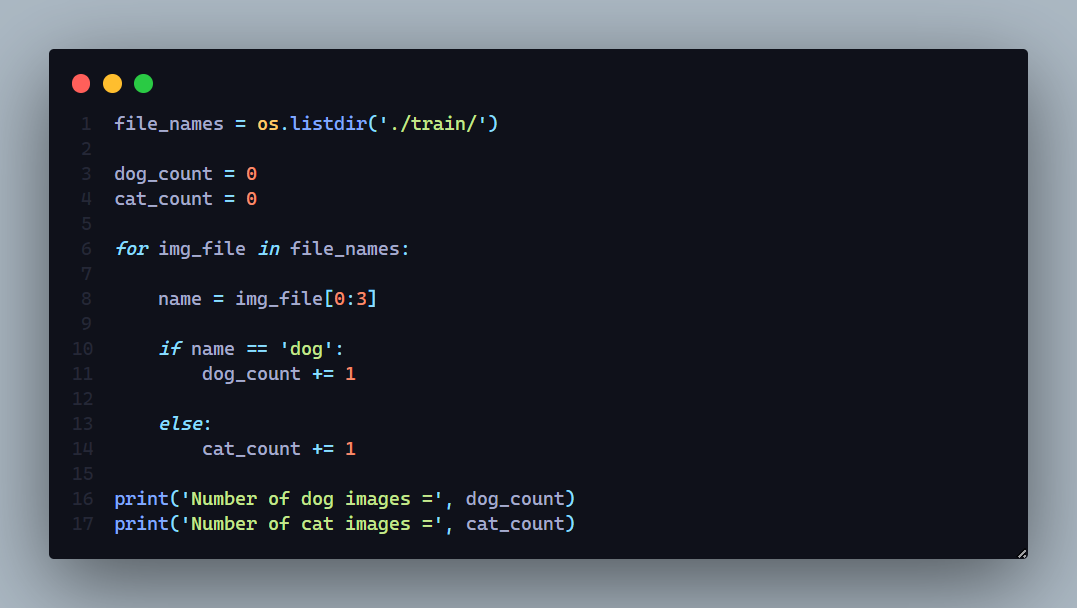
5/8/2023

CS-351L-Artificial Intelligence Lab

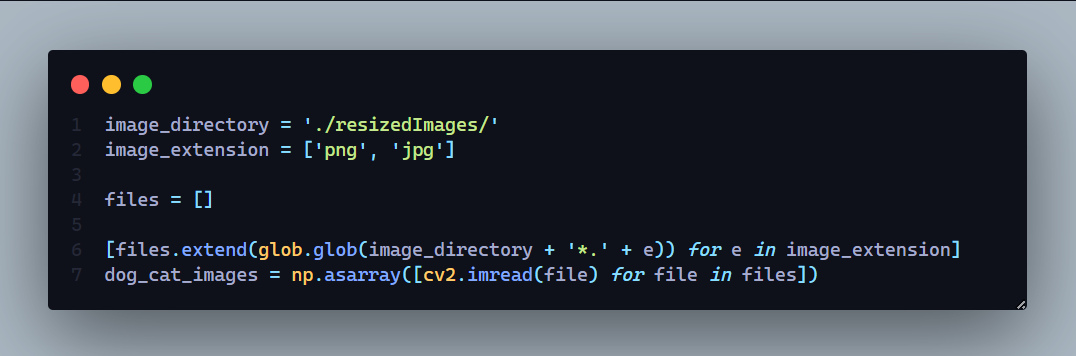
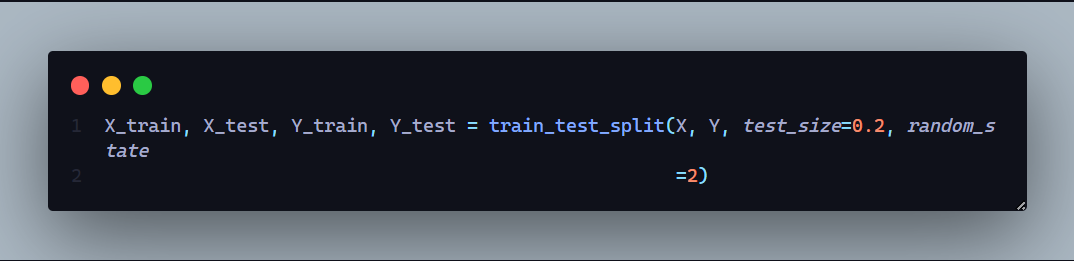
2020244 - Mohsin Zia

Instructor: Professor Muneeb Baig

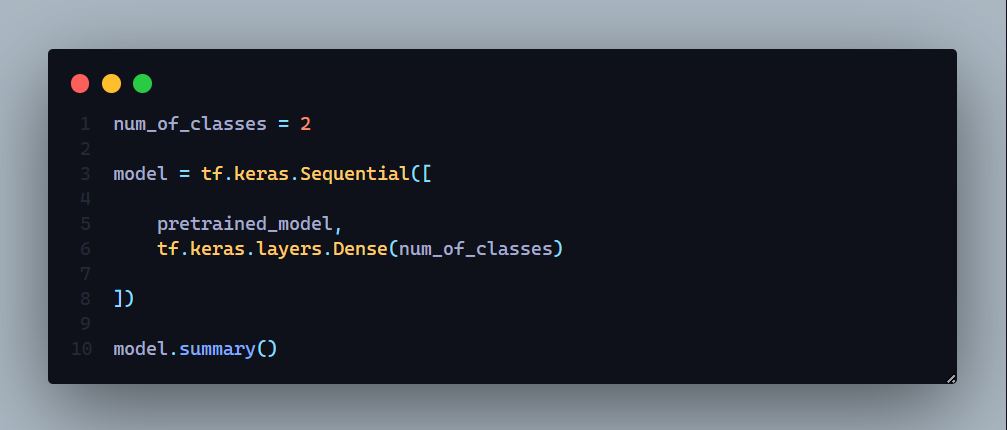
Cats And Dogs Classification using CNN

* First, I’ve imported all the required libraries needed.
* Extracting the zip file which contains the dataset of cats and dogs.
* Saving images from the train folder.
* Counting the images of Cats and Dogs.
* Converting the images to the size that our CNN architecture supports, and then saving them to a new folder.
* A picture containing text, screenshot, software, multimedia

  Description automatically generatedAppending labels for each image to a list. 1 if image contains dog else 0.
* A picture containing text, screenshot, businesscard

  Description automatically generatedCounting the length of each label from the list.
* Type casting the images to 2D Matrix, single pixel values.
* Splitting the data set into train and test set. Test set is 20% of the training data.
* A picture containing text, businesscard, screenshot, font

  Description automatically generatedNormalizing the data, scaling between 0 and 1.
* A picture containing text, screenshot, font, businesscard

  Description automatically generatedImporting the CNN architecture of MobileNetV2 from tensorflow. MobileNet V2 is a family of neural network architectures for efficient on-device image classification and related tasks.
* Assigning the number of classes and creating the CNN model using pretrained model.

|  |  |  |
| --- | --- | --- |
| Layer (type) | Output Shape | Param # |
| keras\_layer (KerasLayer) | (None, 1280) | 2257984 |
| dense (Dense) | (None, 2) | 2562 |
| ===============================================================================  Total params: 2,260,546  Trainable params: 2,562  Non-trainable params: 2,257,984  =============================================================================== | | |

* A picture containing text, screenshot, businesscard, font

  Description automatically generatedCompiling the model.
* A picture containing text, screenshot, businesscard, font

  Description automatically generatedTraining the data on 5 iterations.
* A picture containing text, businesscard, screenshot, font

  Description automatically generatedEvaluating the model on test set and got 95.7% of accuracy.
* A picture containing text, electronics, screenshot, computer

  Description automatically generatedFunction that classifies the image as a cat or a dog. It reads the image from the given path and then displays it. Then the image is reshaped to (1x224x224x3) and then passed to the model for prediction. Then the index with max probability is printed along with it’s class.
* A computer screen shot of a program code

  Description automatically generated with low confidenceFunction that does same as the above function. It just plots the data on subplots.
* Output for cat’s classification on unseen data for the model.
* Output for dog’s classification on unseen data for the model.
* CONCLUSION:

During the completion of this assignment, I have gained profound insights into the remarkable capabilities offered by the CNN architecture known as "MobilenetV2," which is made available through the tensorflow hub. Developed by Google, MobileNet-v2 stands as a 53-layer deep convolutional neural network primarily tailored for training classifiers in the domain of computer vision. The network's utilization of depthwise convolutions facilitates a notable reduction in parameter count when compared to alternative models, consequently culminating in a lightweight deep neural network. By effectively accentuating salient image features, this approach renders classification tasks more accessible for the employed CNN model.

The End.