Capstone Project — Rapolu Pavan Kumar

Machine Learning Engineer Nanodegree — September 17th,2020

Dog Breed Classifier using CNN

Domain Background:

The field of research where this project belongs is computer vision. Computer vision could be defined as the process whereby a machine could get information about some image and transform this data into something intelligible to the human. According to Wikipedia "Computer vision is concerned with the automatic extraction, analysis, and understanding of useful information from a single image or a sequence of images. It involves the development of a theoretical and algorithmic basis to achieve automatic visual understanding."

The Dog breed classifier is a well-known problem in ML. The problem is to identify a breed of dog if a dog image is given as input, if supplied an image of a human, we have to identify the resembling dog breed. The idea is to build a pipeline that can process real-world user-supplied images and identify an estimate of the canine's breed.
This is a multi-class classification problem where we can use supervised machine learning to solve this problem. After completing this model, I am planning to build an app using Tkinter where users can input an image and obtain prediction from this model. This project gives me an opportunity to build and deploy ML models, so I have chosen this as my capstone project.

Problem Statement:

As were stated in the Udacity page the problem will be " to learn how to build a pipeline to process real-world, user-supplied images. Given an image of a dog, your algorithm will identify an estimate of the canine's breed. If supplied an image of a human, the code will identify the resembling dog breed."

Datasets and Inputs:

For this project, the input format is an image type, because we want to input an image and identify the breed of the dog. The dataset for this project is provided by Udacity. The dataset has pictures of dogs and humans.

Dog images dataset: The dog image dataset can be downloaded here https://s3-us-west1.amazonaws.com/udacity-aind/dog-project/dogImages.zip.

The data contains a total of 8351 images which are sorted into a train (6,680 Images), test (836 Images) and valid (835 Images) directories. Each of these directories (train, test, valid) has 133 folders corresponding to dog breeds. The images are of different sizes and different backgrounds, some images are not full-sized. The data is not balanced because the number of images provided for each breed varies. Few have 4 images while some have 8 images.

Human images dataset: The human dataset can be downloaded here https://s3-us-west1.amazonaws.com/udacity-aind/dog-project/lfw.zip

The data contains a total of 13233 human images which are sorted by names of humans (5750 folders). All images are of size 250x250. Images have different backgrounds and different angles. The data is not balanced because we have 1 image for some people and many images for some.

Solution Statement:

For performing this multiclass classification, we can use a Convolutional Neural Network to solve the problem. A Convolutional Neural Network (CNN) is a Deep Learning algorithm that can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image, and be able to differentiate one from the other.

The solution involves three steps:

- First, to detect human images, we can use an existing algorithm like OpenCV's implementation of Haar feature based cascade classifiers.
- Second, to detect dog-images we will use a pretrained VGG16 model.
- Finally, after the image is identified as dog/human, we can pass this image to a CNN which will process the image and predict the breed that matches the best out of 133 breeds.

Benchmark Model:

- The CNN model created from scratch must have an accuracy of at least 10%. This can confirm that the model is working because a random guess will provide a correct answer roughly 1 in 133 times, which corresponds to an accuracy of less than 1%.
- The CNN model created using transfer learning must have an accuracy of 60% and above.

Evaluation Metrics:

For this multi class classification, Multi class log loss will be used to evaluate the model. Because of the imbalance in the dataset, accuracy is not a good indicator here to measure the performance. Log loss takes into account the uncertainty of prediction based on how much it varies from the actual label and this will help in evaluating the model.

Project Design

Step 1: Import the necessary dataset and libraries, Pre-process the data and create train, test and validation dataset. Perform Image augmentation on training data.

Step 2: Detect human faces using OpenCV's implementation of Haar feature based cascade classifiers.

Step 3: Create dog detector using pretrained VGG16 model.

Step 4: Create a CNN to classify dog breeds from scratch, train, validate and test the model. Step 5: Create a CNN to Classify Dog Breeds using Transfer Learning with resnet101 architecture. Train and test the model.

Step 6: Write an algorithm to combine Dog detector and human detector.

- If dog is detected in the image, return the predicted breed.
- If human is detected in the image, return the resembling dog breed.
- If neither is detected, provide output that indicates the error.

References:

1. Original repo for Project - GitHub: https://github.com/udacity/deep-learning-v2-pytorch/blob/master/project-dog-classification/dog_app.ipynb

2. Resnet101:

https://pytorch.org/docs/stable/_modules/torchvision/models/resnet.html#resnet101

3. Imagenet training in Pytorch:
https://github.com/pytorch/examples/tree/master/imagenet

4. Pytorch Documentation: https://pytorch.org/docs/master/

5. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53