UDACITY CAPSTONE PROPOSAL

ARJUN MAINI

Machine Learning Engineer Nanodegree May, 2020

CNN Project : DOG BREED CLASSIFIER

DOMAIN BACKGROUND

This project deals with computer vision which has grown out to be one of hottest areas of research in machine learning today. Due to less amount of datasets, achieving good accuracy has been the biggest problem. Object Identification and image classification are the two crucial steps in this project. Since childhood I have always been fascinated with the different breeds of dogs. Using Convolutional Neural Network (CNN) with Transfer Learning, we will be building a pipeline of layers that can process a real world user supplied image and produce require predictions for us. Through this project, I'd be building this multi-class classification project and deploying a web app for the same.

PROBLEM STATEMENT

We'll be given an image of either a dog or a human. If it is a dog's image, we need to identify the breed of the dog and if it is a human's image, we need to identify which dog breed does the human resemble to. We'll be using supervised learning in this classification problem and transfer learning to speed up the training process and accelerate the rate at which we produce results.

DATASETS AND INPUT

All the data required for this project is provided to us by Udacity. The dataset for dogs is already divided into train, test and valid sets with 133 folders in each set directory of 133 different breeds of dogs. The dataset for humans is also provided sorted by the names. We'll be downloading the data set, unzipping it and then converting the images to NumPy arrays of required pixels.

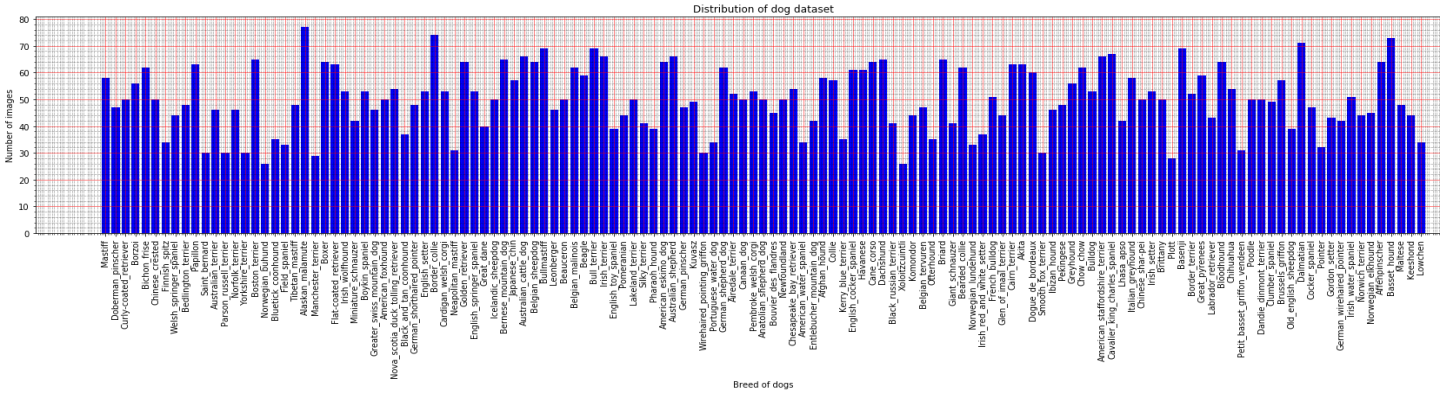
Dog dataset has a total of 8351 images with different backgrounds, from various angles and lighting which is good for the accuracy of our model since we don't know how the provided user input will be. It is also non-uniform i.e. the number of images of different breeds in the folders isn't the same. This dataset has different image sizes so we would resize all to the same size (say 256 x 256).

Dog train dataset : 6680 images
Dog test dataset : 836 images
Dog valid dataset : 835 images

Human dataset has a total of 13233 images (5749 folders) all of the same size but non uniformity i.e. different number of images of different humans. All human images are of same dimensions i.e. 250 x 250.

The input data should be of image type and we want user to supply us the image which would further be sent to our model to obtain the required prediction.

I have plotted the dog train dataset to have an idea about the distribution data :



Here are a few example dog images from our dataset :

INPUT:

OUTPUT:

SOLUTION STATEMENT

For the purpose of solving this multi-class classification problem, first I am using pre trained detectors for detecting whether it's a dog or a human.(VGG-16 for dog detection and OpenCV Haar-cascade for human detection)

I'll be first creating a Convolutional Neural Networks from scratch comprising of convolutional layers followed by max pooling 2D layers (to reduce the dimensional complexity) which will then be converted to scalar and then the scalar will be used to output us the dog breed through SoftMax.

After that, I will be creating a CNN using transfer learning with ResNet-50 and fine tuning it for obtaining predictions with an aim to get an improved accuracy.

BENCHMARK MODEL

We first create a CNN model from scratch. An accuracy of at least 10% would ensure a decent benchmark for the purpose of breed detection given the amount of variability in the data provided. We need to predict 1 out of 133 breeds which is <1% so a 10% accuracy is a good benchmark for the model.

Following is my model architecture:

- 3 convolution (conv2d) layers with kernel size = 3 and stride = 1.
- Max pooling layer of pool_size =2 to reduce input size by 2 times.
- 2 fully connected layers followed by softmax.
- A Dropout layer to prevent overfitting.
- A total of 133 output neurons in the final softmax layer to predict the breed.

EVALUATION METRICS

I would be taking the accuracy of my model as the evaluation metric.
The model created by us should have accuracy of more than 10%.

PROJECT DESIGN

Step 1 : Import datasets and necessary libraries.

Step 2 : Detect human face – we will be using OpenCV implemented Haar – feature cascade classifier to detect human faces.

Step 3 : Detect dog face – we will be using a pre-trained VGG-16 model to detect dog face

Step 4 : Create CNN model from scratch on training set and then validate and test the model

Step 5 : Create CNN model using transfer learning. I will be using a pretrained ResNet-50 model which is a very popular 50-layer deep CNN trained for image classification on millions of images from the ImageNet database. We will fine tune it to detect the 133 dog breeds.

Step 6 : In this step I will construct the main algorithm which takes in an input image, detects whether it is a dog or a human and then matches it with 133 different breeds of dogs.

Step 7 : In this step I will test the created algorithm but supplying an images on my own.

REFERENCES

https://github.com/udacity/deep-learning-v2-pytorch/tree/master/project-dog-classification

https://www.analyticsvidhya.com/blog/2018/12/guide-convolutional-neural-network-cnn/

https://www.mathworks.com/help/deeplearning/ref/resnet50.html