UDACITY'S MACHINE LEARNING NANODEGREE

CAPSTONE PROJECT

DOGS BREED CLASSIFIER USING CNNs

Proposal

Yasir Almutairi

A. Domain Background

Computer vision is the field to make computers recognize and identify objects, such as people, dogs, cars. It might be easy for you or me to identify for a computer it is not a trivial task. There have been many breakthrough in this area by leveraging deep learning in order to classify objects correctly by the use of concepts such as Convolutional neural networks (CNN) which have pushed the capabilities of computers forward and it has many application in self-driving cars, And object classification, And object classification is the domain of the problem I'm trying to solve falls into. In which I will use CNNs in order to identify not only a dog but its breed from 133 breeds.

B. Problem Statement

The objective is to build an algorithm that can accept an image and transforms it to an appropriate format. And classify per conditions given:

- If the image contains a dog then the algorithm have to be able identify the dog, then be able to identify its breed.
- If the image contains a human then the algorithm would return a resembling dog breed.

C. Datasets and Inputs

There are two datasets provided one contains the datasets consist of human image and dog images, both provided by Udacity. Human datasets contains 13,233 images of humans. While dog dataset contains 8,351 images of dogs. To better view the types of breed provided by dog datasets. Look at the two provided figures below.

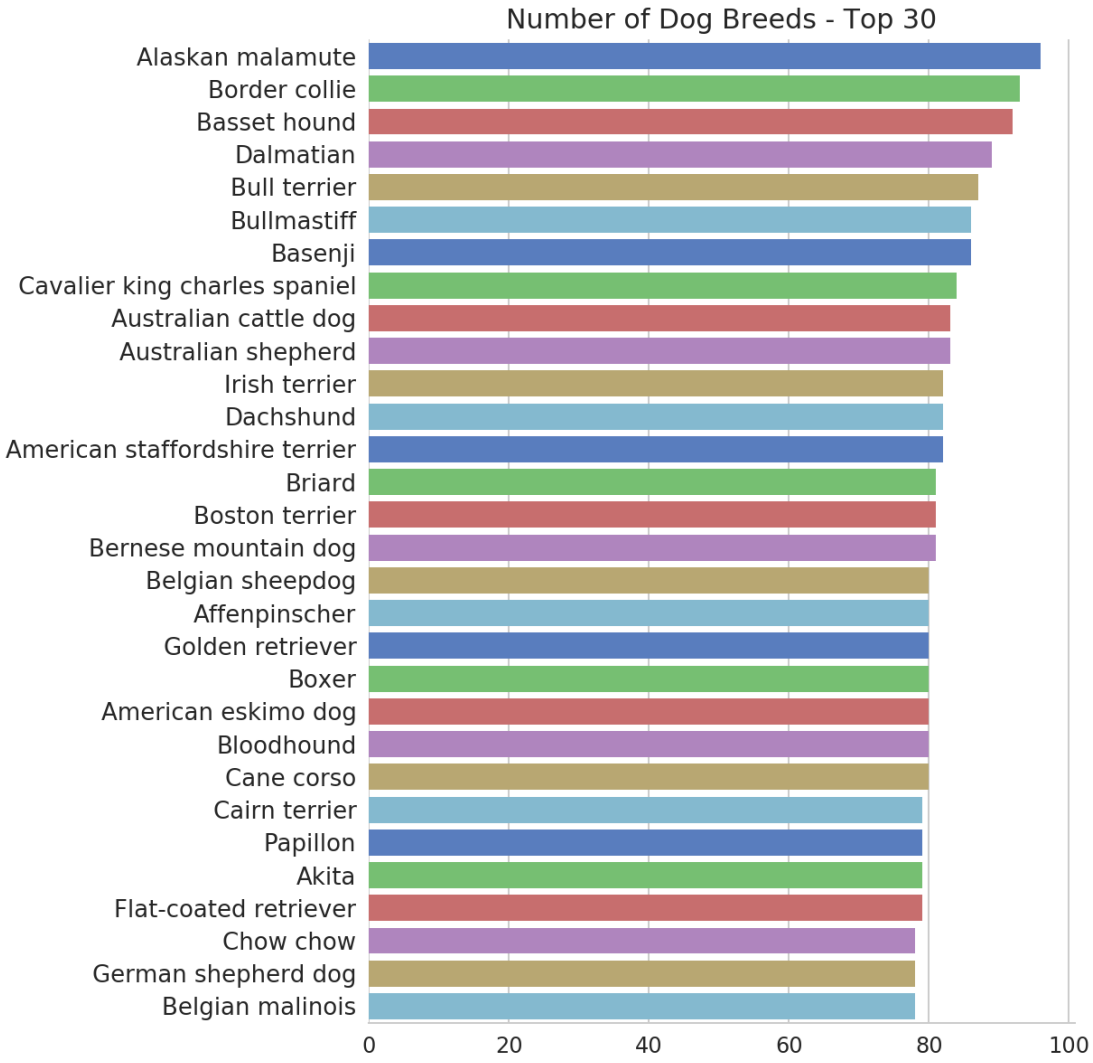

Figure 1: Number of Dog breed

This figure shows the number of top 30 dog breed available in the datasets. To have a better a view of the data the next figure shows the lowest 30 breeds in our data.

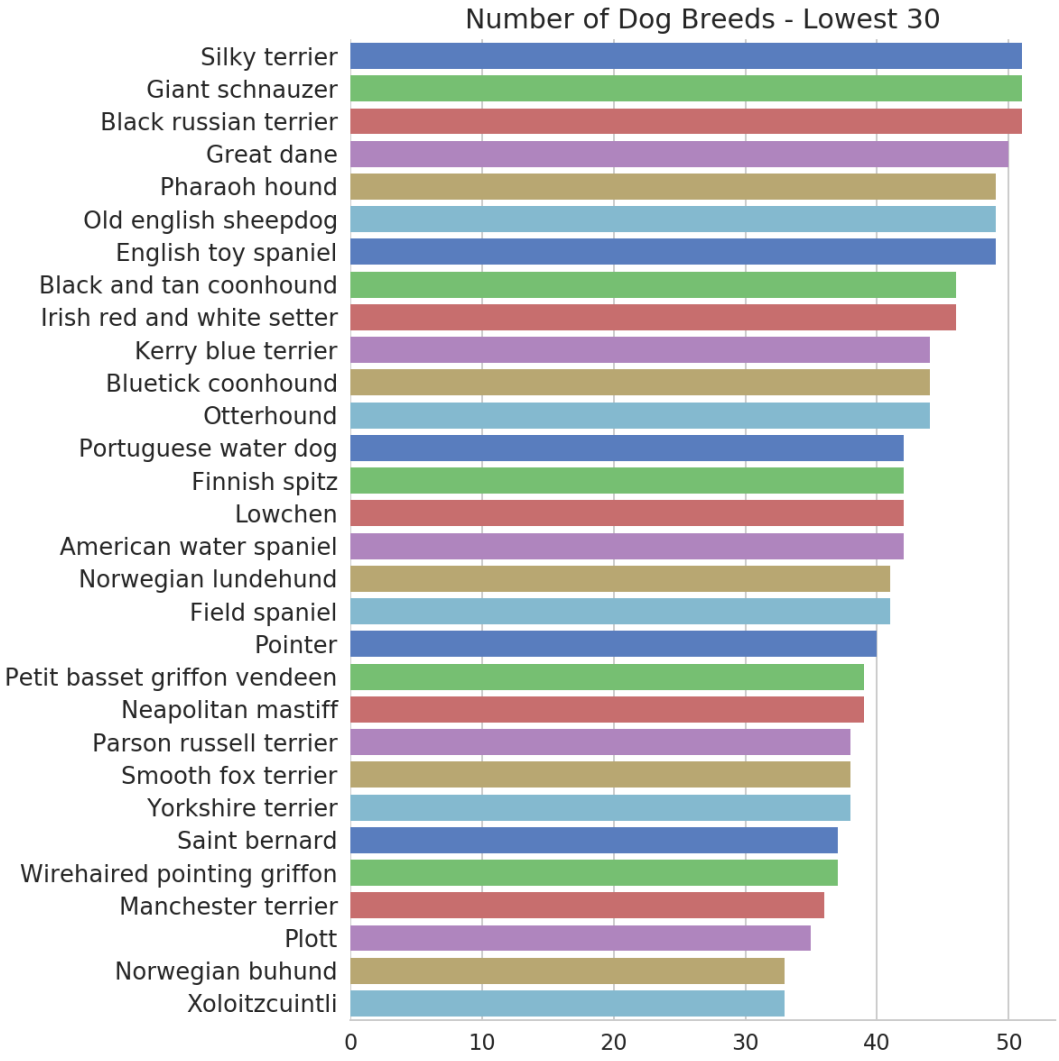

Figure 2: Number of Dog Breeds

what these two figures tells us, Is we have fewer examples for Plott breed than for instance Basenji which might effect the performance of the model in predicting these dog breed that we few examples of them.

D. Solution Statement

The proposed solution is in taking an input image and having the algorithm to detect the presence of a dog and then classify its breed. Also if input image provided contents a human the algorithm would return the closest match of a dog breed. The algorithm would first be implement by

using OpenCV Pre-trained model “Haar Cascades”. And using a pre-trained model from ImageNet competition.

E. Benchmark Model

A benchmark model was built from scratch using CNNs, with minimum test accuracy of 10%. It worth to keep in mind classifying different kinds of breed correctly is not an easy task.

The pre-trained model (using transfer learning) should have a minimum test accuracy of 60%.

F. Evaluation Metric

The metric in which determine the effectiveness of our prediction is accuracy, which would yield the the percentage of correctly classified breeds across all predictions.

$$\text{Accuracy} = \frac{\text{True Positives} + \text{True Negatives}}{\text{All Samples}}$$

Accuracy

G. Project Design

This part will state the workflow of the project in solving this dog breed class.

1. Importing data.

loading the datasets provided by Udacity, and formatting it.

2. Detecting Humans.

In this step we will use OpenCV's pre-trained Haar cascade classifier to detect humans in these images.

3. Detecting Dogs.

We will use the better performing in terms of benchmark. Which is the pre-trained model VGG-16. It has been trained on a large dataset. It dog classification categories is of 1,000 if we look at the sequence from 151, to 268 we will notice they're all dogs. So we can check if the image contains a dog if the category falls between these two categories.

4. Training a Dog breed classifier.

once we able to detect if an image contains a dog using one of models discussed earlier we can proceed in training a dog breed classifier.
by using the pre-trained weights of ResNet-152. Being it trained on large amount of data., it CNN has already learnt useful information through training only the fully connected layers on our data we can have a decent breed classifier.

5. After the model is successful trained with a test accuracy of 75%, we can build our app which would takes an image as input and be able to predict the dog breed, or the resembling dog breed if the image is of human.