Dog-breed Classifier

Simon Andres Soto Ochoa

May 26, 2021

Abstract

This is the proposal for the capstone project for the Machine Learning Engineer Nanodegree from Udacity. In this one, I create the framework that I will use for the development of the project.

1 Introduction

Computer vision is a fascinating area of Computer Science as it helps humans to do jobs at an incredible fast pace. This project in particular is going to work with classification. This is a task that a human can perform but at a slower pace and constraints. In particular, classification of a dog or not, or human or not is not a complicated task for humans but it is more complicated for computers as they only perceive a collection of data specifying the RGB encoding of each pixel. Even though computers have that limitations, there are available several classifiers online to do that in a very accurate way. The project goal is to create the next step, a dog-breed classifier. This is not a trivial task, as there are many several breed that look quite alike. For a human it requires a lot of time just to memorize the name of the breed, something that a computer can do better than us, but still, after learning names, the process of choosing correctly a specific breed is another story. This has been a topic in research, for example [LKJB12] , tries to overcome difficulties by identifying more features for a breed in the same pictures. Extracting more features from the image has shown results in the case of flowers [NZ08]. There are attempts only in the area of feature extraction to find missing dogs [LTY19].

Besides this topic there are more interesting projects to conduct, but as this project goes well, will give a model from which I can base other personal projects.

2 Problem Statement

Classification of dog breeds is a complex topic even for humans. Therefore, we are going to construct an algorithm that can help in that process. Given a picture, determine if the picture contains a human of a dog. In the later case, determine a dog breed, and in the other case, determine the breed that the human resembles.

Restrictions: Picture must have ONLY a dog or a human. First part consists in identifying if there is a dog or a human, and the second part is to return the most likely breed or the breed that the human resemble the most. If both or none of the previous ones is found in the picture, we return an error, and ask for a new image.

Ideas of Solution: Use *Transfer Learning* from a Convolutional Neural Network (CNN) to be able to construct (not from scracht) without training our own architecture a dog classifier. We are going to define the architecture of the network but we will use a pre-trained model.

3 Data Sets and Inputs

The data sets are publicly available. Here you can download the .zip file for the dogs pictures and here for the .zip file containing the pictures of humans.

The dog folder contains the train, validation, and test sets already in different folders in which each folder has a respective dog breed. This follows the regular standards on image classification in which each example has to be well referenced. Similar for humans, but in this case, we have just a folder

named after one person and inside at least one picture of him/her. There are 8351 images of dogs and 13233 images of humans.

In our case, we have 133 different breeds (The number is the same for train, validation and test) a pre-trained model (VGG-16), and a model to identify human faces in pictures from OpenCV.

4 Solution Statement

Use *Transfer Learning* with a model that has been trained for image recognition.

5 Evaluation

As we saw in Section 3, for each breed, we have pictures in the train, validation, and test sets. The classes are no balanced but also not completely imbalance as it can be appreciated in figure 1. There is a difference among the top 4 breeds with more images and the bottom 4, but this does not make up for a significant difference. Therefore recall (True Positive Rate) is a good metric for this problem. In section 6, recall is the measure that is used in the benchmark model, which also helped in the decision to go with it here.

We are not going to check or think about the identification of a human or a dog in a picture. These parts are going to be solved using already existing algorithms (OpenCV and VGG-16).

6 Benchmark model

The benchmark model is going to be a trained neural network design by us. In this one, we are going to take ideas from different sources, for example [LKJB12], which is one of the first papers that classifies the 133 dog breeds (same number as in this project). In this paper, they use the recall (True Positive Rate) to measure the performance of the model.

7 Project Outline

Following the order found in the example jupyter notebook to follow. This is organized in the following way:

1. Detect humans: Use of the openCV haarcascadefrontal classifier.
2. Detect dogs: Use the pre-trained VGG-16 model from pytorch to detect dogs in pictures.
3. First Approach: Construct a Neural Network to classify dog breeds.
4. Second Approach: Use of transfer learning to improve the results obtained previously.

At the end of those, validate and test the final architecture with the transfer learning. Show some example on how it works on at small sample from both categories (humans and dogs).

References

[LKJB12] Jiongxin Liu, Angjoo Kanazawa, David Jacobs, and Peter Belhumeur. Dog breed classification using part localization. In *European conference on computer vision*, pages 172–185. Springer, 2012.

[LTY19] Kenneth Lai, Xinyuan Tu, and Svetlana Yanushkevich. Dog identification using soft biometrics and neural networks. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

[NZ08] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008.

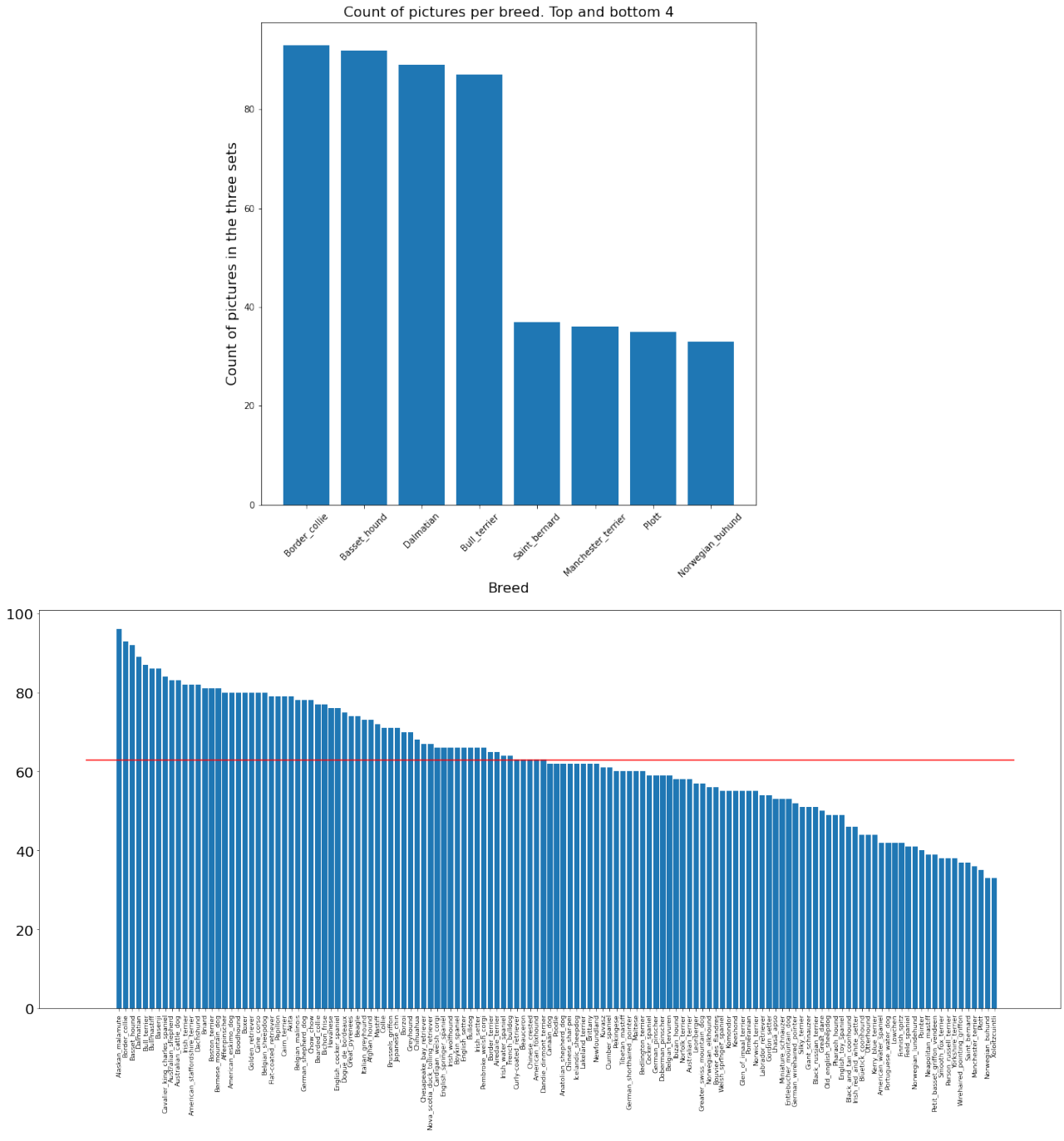

Figure 1: Number of pictures per breed adding up the three sets.