Dog breed Classifier

Machine Learning Engineer Nanodegree

Capstone Proposal

Fernando de la Torre

February 20th, 2021

Domain Background

In 2012, a deep convolutional neural network architecture capable of outperforming the previous results on a difficult dataset such as imagenet, was released. Alexnet and the paper that is considered one of the most influencing in computer vision were born [1]. Only in 2020, the AlexNet paper has been cited over 70,000 times according to Google Scholar [2].

Since then, there have been more and more effective models for image classification, such as VGG16, GoogleNet and Resnet. All of them have given the opportunity to use computer vision in different areas: medical diagnosis, social networks, self driven cars, surveillance security cameras, face recognition… Today, these convolutional neural networks are everywhere.
The aim of this project is to use some of these deep learning models to classify different dog breeds´ images.

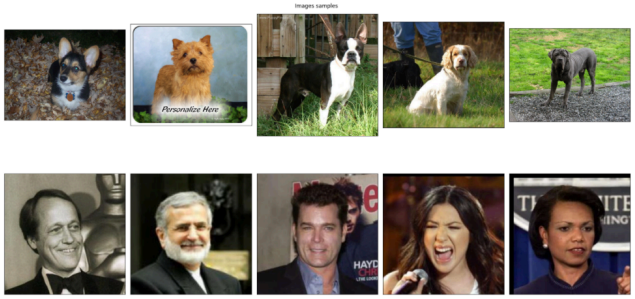
Problem Statement

American Kennel Club [3], the largest purebred registry in the world, currently lists 190 dog breeds in the United States alone [4]. Even for someone who knows about different kinds of dogs, it can sometimes be hard to tell the exact breed of a dog.

Convolutional neural networks (as Alexnet) are a class of deep neural networks most commonly applied to computer vision. With the help of CNNs, we will create a classifier capable of, firstly, identifying a dog or a person in an image. Then, if there is a dog, it will classify its breed among 133 types. If there is a person in the image, the model will also come up with a dog breed; the one closest to the given face features.

Datasets and Inputs

In this project, we will be using the proposed dataset provided by Udacity for their Machine Learning Engineer nanodegree. It is composed by 8351 dogs´ images and 13233 human faces.



Dataset samples

The dog images are divided into 6680 for training, 835 for validation and 836 for testing. There are the same 133 breeds in each set: from 26 to 77 images of each in training set and from 3 to 10 in validation and testing sets. The images themselves have different backgrounds, lights and the dog can appear in any pose and size. This is convenient for our model, because it can learn a lot of possibilities of what a dog is in a picture. This way we can assure that, later, it will be capable of recognizing a dog in any given image, no matter the angle or the light. The size of the images are different, too.

The human dataset contains images of 5749 different people and from 2 to 530 images for each. All of them have the same size.

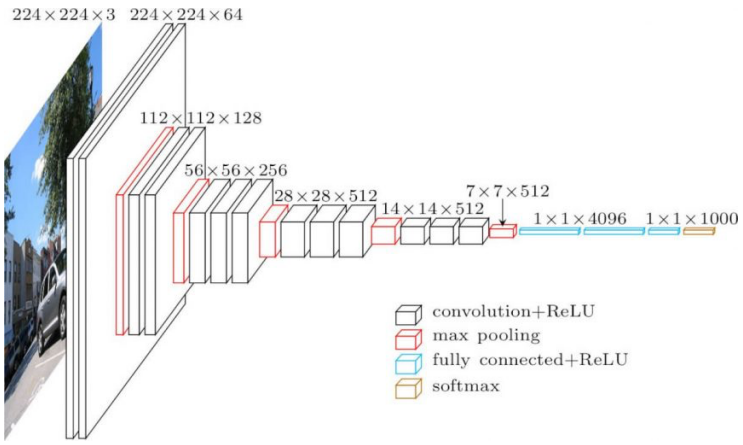
Solution Statement

The purpose of the project is to be able to recognise a dog breed in an image with the best possible results. To achieve this, we will use different CNN models : pre-trained, built from scratch and lastly, using transfer learning. For the face recognition part we will use a Haar feature-based cascade classifier, too.

The last step is to write an algorithm that can detect a dog or a human in an image and recognise the dog breed or the resembling one if it is human.

Benchmark Model

We will use the pre trained model VGG16, which achieves 92.7% top-5 test accuracy in ImageNet, a dataset of over 14 million images belonging to 1000 classes. [5]



VGG-16 architecture

We expect that this model will return indexes that correspond to dog breed classification if a dog is recognized in a picture.

Evaluation Metrics

There is a popular Kaggle competition of dog breed identification, such as this project. They use multi-class log loss metrics for evaluating the results of the contestants. Log loss is similar to the accuracy metric, but it will favor models that distinguish more strongly the classes. It is useful to compare models not only based on their output but also on their probabilistic outcome [6].

$$LogLoss = \frac{-1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} x_{ij} * \ log(p_{ij})$$

Logloss for multi-class classification [7]

We will use multi-class log loss metrics to evaluate our results too, in order to compare results with the Kaggle public leaderboard. The mean loss score is 2.1695, so a reasonable score will be under this value. The closer to 0, the better.

We will use accuracy metric too as a second measure.

Project Design

This project will develop following these steps:

1. **Data preparation and analysis.**

The first task will be focused on doing a preliminary exploration and analysis of the given data. We will evaluate the balance, the need of data augmentation and the images themselves so we are able to use them correctly later.

2. **Detect human faces.**

We will use Opencv implementation of Haar feature cascade classifier.

3. **Detect dogs with a pre-trained model.**

VGG-16 will be used to detect dogs in our images. With this model and our face detector built in step 2, now we are capable of discerning a person and a dog in an image.

4. **Create a dog breed classifier from scratch.**

We will create a CNN model and train it with the dataset. Later, we will test it with the aim of getting an accuracy over 10%.

5. **Create another CNN dog breed classifier using transfer learning.**

The result of the model is expected to overcome the results of the previous model, because we take advantage of the pre-trained weights and custom the model to our needs. This time, the goal is to obtain al least 60% of accuracy on the test set.

6. **Develop a final algorithm.**

With the previous human detector from step 2 and the dog breed classifier from step 5, we finally write an algorithm that takes an image in a given path and and determines if it contains a human or a dog. If there is a dog, it will predict its breed and if there is a person, the resembling dog breed.

References

[1] https://papers.nips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

[2] https://en.wikipedia.org/wiki/AlexNet

[3] https://www.akc.org/press-center/articles-resources/facts-and-stats/

[4] https://www.hillspet.com/dog-care/behavior-appearance/how-many-dog-breeds-are-there

[5] https://neurohive.io/en/popular-networks/vgg16/

[6] https://medium.com/@fzammito/whats-considered-a-good-log-loss-in-machine-learning-a529d400632d

[7] https://medium.com/analytics-vidhya/performance-metrics-for-machine-learning-models-80d7666b432e