Dog Breeds Classifier (CNN)

Domain Background

This project is based in the field of Computer Vision. Computer vision involves using a computer to make classification, identification, and/or inference from visual data. Such data can include but it not limited to, images, videos, and footage. Currently, the best way to implement vision is to remap visual data into numerical data, such as color intensities and color values. This remap enables deployment of traditional learning algorithm onto the data. In my line of work, I often come across data such as scans of legal documents, drug labels, and handwritten data. Such data requires techniques such as OCR, classification, and image identification that all fall under the field of Computer Vision. There also many other applications of this such as facial recognition, barcode scanning, automatic check deposits at ATMs, etc.

Problem Statement

The goal of this project is to develop a ML model that can identify images of dogs and guess their breed. In addition to this, the model will also be able to identify images of humans, and guess a dog breed that they most resemble. As dogs generally fall into well defined breeds and are visually distinct from humans, there is a clear set of criteria for benchmarking the performance of this algorithm.

Datasets and Inputs

The input dataset consists of about 18K images of humans and and 8K images of dogs, separated into folders by breed. Most of these images are medium sized (10-500 kilobytes) and are of good enough quality for a human viewer to identify the content. The dog images were obtained from an S3 bucket hosted here. The human images were obtained from a public data store hosted by University of Massachusetts here. The datasets will be used in both the training and testing of the ML model to be used for identification. Additionally, the pre-trained models (described in **Benchmark Models** below) contain over a million images obtained from various URLs on the Internet.

Solution Statement

The solution to the problem will be to develop a Convolution Neural Network (CNN) model to identify and classify the content of the images. The solution will accept images as input from a user (or possibly from a web app, smartphone app, or any other API) and produce the following output:

1. If the image is identified as a human, the program will greet the human (“hello, human”), and then proceed to suggest a dog breed most similar to the person’s likeness.
2. If the image is identified as a dog, the program will greet the dog (“hello, dog”) and then proceed to identify the breed of the dog.

Benchmark Models

There exist pre-trained models such as the OpenCV's Haar feature based classifier used to identify human faces. There is also VGG-16 model that contains over 1000 different classification categories with as many as 118 categories for dogs. The VGG model is trained using Deep Convolution Networks, which is a more advanced implementation of the CNN implemented by the project. These will be used as benchmark models for comparing the CNN model that I will develop. In addition, the project will use these models to give the CNN model a leg-up by using Transfer Learning.

Evaluation Metrics

The main evaluation metrics here will be accuracy. In other words what proportion of supplied images were correctly identified and classified. There will be subcategories of accuracy such as identification of human vs dog and identification of dog breed for a dog image. There is no effective measure for guessing similarity between a human and a dog breed as there is no clearly defined correct answer. A visual review will be the only metric used for this prediction.

In addition to this false positive, false negative, recall, and precision rates can also be looked at to review the performance of the human vs dog identification.

The from-scratch model is expected to perform at 10% accuracy, while the Transfer Learning model is expected to reach 60%.

Project Design

The project will first involve using and testing standard algorithms like the Haar and VGG-16. As the dog image dataset is already pre-classified into train, test, and validation datasets, this is easy to do. The next step will be to design a CNN from scratch. This will involve many preliminary steps such as converting the images to grayscale and resizing to make the dataset uniform. The effect of decisions made regarding these conversions will probably impact the outcome and thus will be varied to find an ideal specification. After training the initial CNN model from scratch the project will proceed to implement a more standard workflow of training a CNN by transfer learning from the above standard algorithms. When training the project will make several implementation decisions such as the architecture, number of layers, loss functions, optimizer functions etc. These will all affect the ultimate outcome and will thus need to be carefully optimized. Depending on the speed of the computation, it may be necessary to requisition ML instances on Amazon Sagemaker to complete the training and prediction jobs.