Convolutional Neural Networks (CNN) project
Dog Breed Classifier

Stephen Parvaresh
January 2021

Domain Background

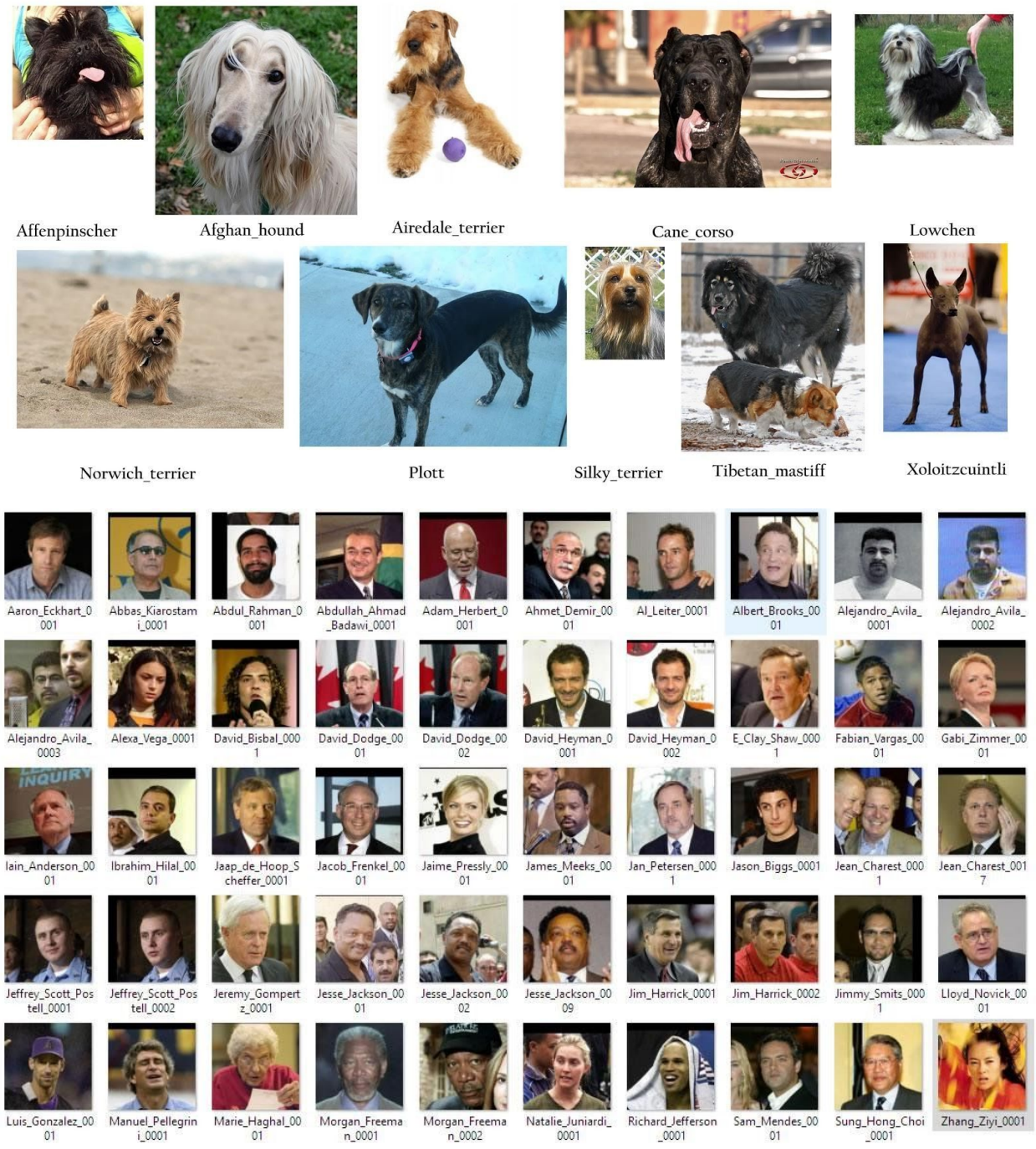
Machine learning (ML) has become more popular thanks to very powerful computers that can handle processing large amounts of data in a smaller amount of time. The concept of machine learning was introduced by Arthur Samuel in 1959, but today it is reaching its full potential with better processing ability and the availability of large amounts of data.
The dog breed classification problem is well known in Machine Learning. We can find it on Kaggle where it was the topic of a competition in 2017.
Further, the idea of classification has been used in academic work to identify different types of flowers.

Problem Statement

The purpose of this project is to build a convolutional neural network that takes images of dogs or humans as inputs and returns whether the image is of a dog or human, and what dog breed it is. This is a supervised learning problem and, because the dog images are divided into breed classes, classification predictive modeling will be utilized.

Datasets and Inputs

To solve the problem, the input data must be images of either dogs or humans. All data for this project is provided by Udacity, and contains pictures of dogs and pictures of humans.
All dog pictures are sorted into train (6,680 images), test (836 images) and valid (835 images) datasets, and all the images in those datasets are sorted into breed classes. Therefore, there are a total of 133 folders (dog breeds) in every train, test and valid directory.
Human pictures are sorted by the name of each human, totaling 13,234 images of 5,749 humans. The data is imbalanced because not all humans or dog breeds are represented evenly.
Dog images have different resolution sizes, backgrounds, lighting, and style (some dogs are shown in full, while others are just the face). This variety of images is good because it will allow our model to train on a variety of test cases, as the images of humans are not standardized.
Below are a few samples of dog and human images:



Solution Statement

The model will be a Convolutional Neural Networks (CNN), which is a part of deep neural networks and is great for analyzing images. To determine if the picture is of a human or not, the OpenCV model will be used. To determine if the image is of a dog, a pre-trained VGG16 model

will be used. Then, the CNN model will be created using transfer learning, which uses fewer images with similar results.

Benchmark Model

For the benchmark model, a Convolutional Neural Networks (CNN) model created from scratch will be used with an accuracy of more than 10%. This should be enough to confirm that the model is working because random guess would be 1 in 133 breeds, or less than 1%.

Evaluation Metrics

The problem to be solved is a classification problem. Because the data is imbalanced, a simple accuracy score is not a useful metric to determine model performance. On Kaggle, the multi-class log loss metric was used to evaluate models. For this project, that metric will also be used in order to compare performance to the results on Kaggle. F1 score testing will also be used because it considers precision and recall.

Project Design

The first thing to be modeled is whether images of humans can be detected. To do this, the OpenCV model will be used to get faces from the images, which will identify the image as containing a human or not. The workflow on detecting faces is as follows:

– initialize pre-trained face detector
-load image
-convert image to grayscale
-find faces in the image
-return true if the number of faces is more than 0 else return false

Then the model will detect dogs in images. The pre-trained model VGG16 will be used for this.

– the VGG16 model is defined
– load and pre-process the image
– send an image to the VGG16 model
– model return index from 0 to 999 (dog classes are from 151 to 268)
– return true if the index is >=158 and <=268 else return false

The data is already divided into training, validation and test partitions so the train dataset can be used to make a benchmark model using Convolutional Neural Networks. After creating a model, the test data will be used to test the model. When an accuracy score of over 10% is achieved, a new model using transfer learning will be built. With transfer learning, a model can

be built using fewer images to obtain similar results. The same training dataset will be used as before. The model will then be tested on the same test dataset, though an accuracy score of over 60% will be expected. Then different model parameters will be utilized to get better results. Finally, an F1 score and log loss will be used to evaluate the models.

Reference

https://en.wikipedia.org/wiki/Convolutional_neural_network
https://www.kaggle.com/c/dog-breed-identification/overview/description
https://github.com/udacity/dog-project
https://www.kdnuggets.com/2018/04/right-metric-evaluating-machine-learning-models-1.html
https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234