Capstone Project – Dog breed classifier

Machine Learning Engineer Nanodegree

Avani Jain Patel

9th Sep 2020

1. Project Definition

1.1 Project Overview

Computer Vision has played very important role in solving critical problems in different industries like automobile, health care, security services etc. One of the well-known examples for image classification is dog breed classification. One can classify dog breeds based on characteristics like color, hair, shape of tail or face, however with breeds which are very close in terms of these characteristics there is need for automation. Machine learning supervised learning comes in play in such scenario. Many times, when people who are looking to get dog as pet might want to know breed of dog. e.g. Labradors come in yellow, chocolate, and black with machine learning we should be able to identify this intra-class variation.

My interest in selecting this project is to learn about image classification using CNN. Coming from data engineering background this project would give me opportunity to learn to create model, test and validate its accuracy. Use various techniques, pre trained models like ResNet50, VGG16. This pipeline will accept the image of dog and identify its breed.

1.2 Problem Statement

Goal of this project is to estimate dogs breed for the input image provided. If the human image is provided instead dog it should estimate closest matching dog breed for human face. Accuracy of 60% or greater should be achieved. To solve this problem CNN model with transfer learning is used.

1.3 Metrics

For this project I will compare performance of my model with benchmark model. Therefore accuracy. Is used as an evaluation metrics. CNN model created from transfer learning should have accuracy of 60% or above.

2. Analysis

2.1 Data Exploration

Datasets for this project is image since we are classifying dogs breed based on input image. Images are provided BY Udacity which includes both human and dog images.

Dog dataset-

There are total 8531 dog images which is split into train, test and valid folders. Images are of different size & backgrounds. Total 133 dog breed images are provided which includes from Afghan to Yorkshire terrier. Number of images provided varies for each breed in each folder, some have 4 images and others have 8 images There are 836 test files, 6680 training files and 835 validation files. Below is one such example image of Belgian Malinois breed. All images are in .jpg with different resolution.

Human dataset-

Human dataset contains 13233 image files. These images are for normal human images. It is sorted by names of human. (5750 folders). These images have different backgrounds and alignments. Data is not balanced as there more images for some people as compared to other people.

2.2 Algorithms and Techniques

To perform multiclass image classification, CNN (Convolutional Neural Network) is one of the best or most popular technique in computer vision. There are 3 steps in project. First step is to detect human images. For this purpose, I have used OpenCV’s haar feature based cascade classifier. 2nd step is to identify dog images. For this I have used pre-trained VGG16 model. This is trained model on ImageNet dataset. Finally, once image is identified as dog/human this image is passed to CNN model to predict the closest matching breed out of 133 breeds. CNN model is built from scratch to classify dog breeds. CNN model is built using Transfer learning to classify dog breeds. ResNet50 pre-trained model is used. Model is trained with dog dataset and fined tuned.

2.3 Benchmark

The CNN model created from scratch with accuracy of more than 10% will be used for benchmark. This should be enough because random guess would be 1 in 133 which less than 1% are.
CNN model created from transfer learning should have accuracy of 60% or above. For this project I will compare performance of my model with benchmark model. Therefore accuracy is used as an evaluation metrics.

3. Methodology

3.1 Data Preprocessing

Not all images are of same resolution and equally balanced. Grey scaled images are converted to multiscale to detect human faces. Dog images are converted to 244 X 244 to be used by VGG16 model.

All trained images are RandomResizedCrop & then RandomHorizontalFlip is performed.

All validation set data is reduced to 256X256 pixels and then center cropped to 224X224 images.

All test set data is resized to 224X224 and normalized.

3.2 Implementation

1. Import dataset. Dataset was available on Udacity. I downloaded data and then uploaded to S3. It was then downloaded from S3. Datasets mentioned above is used.
2. As step 2, I have created function to detect human images. For this purpose OpenCV model provided by Haar Cascade is used. haarcascade_frontalface_alt.xml model is used.
3. Then I have used pre-trained VGG16 model to detect dog images. This is trained model on ImageNet dataset.
4. Create CNN model from scratch to classify dog breeds. Then will train, validate and test the model.
Model has 3 convolution layer.
For first 2 convolution layer kernel size of 3 with stride 2 is used which will lead to downsize of input image by 2.
After 2 conv layers, maxpooling with stride 2 is placed and this will lead to downsize of input image by 2.
The 3rd conv layers is consist of kernel size of 3 with stride 1, and this will not reduce input image.
After final maxpooling with stride 2, the total output image size is downsized by factor of 32 and the depth will be 128.
Here dropout of 0.3 is applied in order to prevent overfitting. Fully-connected layer is placed and then, 2nd fully-connected layer is intended to produce final output size which predicts classes of breeds.

3.3 Refinement

Model created from scratch provided the accuracy of 9%. Though it's not meeting benchmarking the model can be significantly improved by using transfer learning. Also as this is my first experience working on ML project and creating model I decided to proceed further to refinement of the model.

For this purpose I have used ResNet50 pre-trained model. Model is trained with dog dataset and fined tuned. With this accuracy achieved was 70% which is greater than 60%.

4. Results

4.1 Model Evaluation and Validation

Human face detector function 99% of accuracy (99 images detected in 100 images) using haar cascade model.14% of error was detected as detecting humans in dog images.

Dog detector was able to detect 88 images with dogs in first 100 dog image input. No dog were detected in human images.

CNN model built from scratch and trained up to 10 epoch produced accuracy of 9%.However with RestNet50 I was able to fine tune the model to give 70% accuracy by training for 20 epoch.

4.2 Justification

I feel for the first time implementation and limited knowledge on CNN model techniques the accuracy achieved of 70% as compared to base model of 9% is much better. As I explored more I think ResNet101 would have better performed then ResNet50. May be this is something I need to explore more and understand the concepts for my future projects.

4.3 Improvement

I have tried only ResNet50 but I think there might be other better techniques or models which will outperform. I would also like to add the API with AWS API Gateway/Lambda & S3 where you can provide image as input through API for future projects.