Machine Learning Nanodegree Capstone

Project Proposal: Dog Break Classifier

Deepak Gupta

19th April, 2020

1 Domain Background

Recognising dogs according to their respective breed is a challenging task even for humans. There are hundreds of breeds in existence which are grouped into 10 distinct groups (Group 1-10) according to physicial characteristics. This shows that classification is a complex task as remembering all the breeds and distinguishing between similar breeds cannot be easily done by humans. Therefore, the need for this to be done by a computer is necessary. The task at hand involves image recognition and classification which uses machine learning tech-niques to aid computer vision. A convolutional neural network (CNN) will be used to accomplish this task.

2 Problem Statement

The aim of the project is to build a pipeline to process real-world, user-supplied images. The algorithm will identify an estimate of the dog's breed given an image. When the image is of a human, the algorithm will choose an estimate of a dog breed that resembles the human. If neither a dog or a human is detected, then an error message is output. Therefore, the models in place should be capable of detecting a dog or human in an image, classify the dog to its breed and classify a dog breed that the human resembles.

3 Datasets and Inputs

All the datasets used to tran, test and validate the CNN model provided by Udacity. There is the dog dataset containing 133 breeds and each breed contains 8 images. There is also the human dataset which contains images, first names and last names. There is a total of 13233 human images and 8351 dog images.

4 Solution statement

For performing this multiclass classification, we can use Convolutional Neural Network to solve the problem. A Convolutional Neural Network (CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. the project will first use OpenCV's implementation of Haar feature-based cas- cade classifiers to detect human faces from the images supplied. Next, a pre- trained VGG-16 model with trained weights on ImageNet, a large popular dataset for image classification, will be used to detect dogs in the user-supplied images.

5 Benchmark model

A CNN model is then built from scratch to classify dog breeds, that is, transfer learning cannot be used just yet. This model should surpass a test accuracy of 10% set by Udacity because the model is being built from scratch so classifying similar breeds can be a challenge however transfer learning will greatly improve this. Finally, a transfer learning will be used with a ResNet50 model to significantly boost the accuracy of the CNN model. It should surpass the 60% test accuracy set by Udacity.

6 Evaluation Metrics

Multi class log loss will be used to evaluate the model. Because of the imbalance in the dataset, accuracy is a not a good indicator here to measure the performance. Log loss takes into the account of uncertainty ofprediction based on how much it varies from actual label and this will help in evaluating the model.

7 Project Design

The workflow for the project to approach the solution is given below:
Step 0: The datasets for the human and dog images are imported and the total number of dog and human images is determined.
Step 1: OpenCV's implementation of Haar feature-based cascade classifiers is used to detect human faces from the datasets.
Step 2: Pre-trained VGG-16 model is used to detect dog images from the datasets.
Step 3: CNN model is created from scratch to classify the images. Test accuracy should be greater than 10%. Resizing and cropping the image. Augumentation through rotations.

Step 4: CNN model is created using transfer learning to classify the images. Test accuracy should be greater than 60%. Resizing and cropping the image. Augumentation through rotations. CNN layers are utilized for feature extraction and image generalization. The dropout layer ensures that overfitting is dealt with and the linear classifier converts the extracted features to a classified type.
Step 5: An algorithm is developed that returns the predicted breed if a dog is detected. If a human is detected, it should return the resembling dog breed. . If neither dog or human is detected, then error message is output.
Step 6: Testing the solution model with images from the datasets to see the model in work.

7 References

1. Original Project Repo. : https://github.com/udacity/deep-learning-v2-pytorch/blob/master/project-dog-classification/
2. Resnet101: https://pytorch.org/docs/stable/_modules/torchvision/models/resnet.html#resnet101
3.Udacity Dataset: https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/dogImages.zip