Capstone Project Purvansh Singh

Machine Learning Engineer Nanodegree July 3rd, 2020

Dog Breed Classifier using CNN

Project Overview

Breed classifier is a well-known Machine Learning Problem, The problem consist of identifying the breed of dogs if dog image is given as input, if we supply image of a human then we have to identify the resembling dog breed. We have to build a model that can process real world user supplied images and identify an estimate of the canine's breed. This is a multi-class classification problem where we can use supervised machine learning to solve this problem, and it is a multi-class classification problem where we can use supervised machine learning to solve this problem.

Problem Statement

project is all about building a machine learning model that can be used within web app to process real-world, user-supplied images. The algorithm has to perform two tasks:

- ***Dog face detector***
- ***Human face detector***

Metrics

Dataset is split into train, test and validation set. The model is trained using the train dataset. We use the testing data to predict the performance of the model on unseen data. We will use accuracy as a metric to evaluate our model on test data.

Accuracy=Number of items correctly classified/ All classified items

Also, during model training, we compare the test data prediction with validation dataset and calculate Multi class log loss to find the best performing model. Log

loss takes into the account of uncertainty of prediction based on how much it varies from actual label and this will help in evaluating the model.

Data Exploration

For this project, the input format must be of image type, because we want to input an image and identify the breed of the dog. The dataset has pictures of dogs and humans.

Dog images dataset: This Data set contain 8351 total images which are sorted into train (6,680 Images), test (836 Images) and valid (835 Images) directories. Each of this directory (train, test, valid) have 133 folders corresponding to dog breeds. The images are of different sizes and different backgrounds, some images are not full-sized. The data is not balanced because the number of images provided for each breed varies. Few have 4 images while some have 8 images.

Human images dataset: This dataset contains 13233 total human images which are sorted by names of human (5750 folders). All images are of size 250x250. Images have different background and different angles. The data is not balanced because we have 1 image for some people and many images for some.

Sample Images of the dataset

Algorithms and techniques:

For this multiclass classification problem, we can use Convolutional Neural Network to solve the problem. A "Convolutional Neural Network" 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 solution involves three steps.

1. First, to detect human images, we can use existing algorithm like OpenCV's implementation of Haar feature based cascade classifiers.
2. Second, to detect dog-images we will use a pretrained VGG16 model.
3. Finally, after the image is identified as dog/human, we can pass this image to an CNN model which will process the image and predict the breed that matches the best out of 133 breeds.

Benchmark

The CNN model created from scratch must have accuracy of at least 10%. This can confirm that the model is working because a random guess will provide a correct answer roughly 1 in 133 times, which corresponds to an accuracy of less than 1%.

Data Preprocessing

Images are resized to 224*224, then normalization is applied to all images (train, valid and test datasets). For the training data, Image augmentation is done to reduce overfitting. The train data images are randomly rotated and random horizontal flip is applied. Finally, all the images are converted into tensor before passing into the model.

Model Evaluation and Validation

Human Face detector: The human face detector function was created using OpenCV's implementation of Haar feature based cascade classifiers. 98% of human faces were detected in first 100 images of human face dataset and 17%

of human faces detected in first 100 images of dog dataset.

Dog Face detector: The dog detector function was created using pre-trained VGG16 model. 96.0% of dog faces were detected in first 100 images of dog dataset and 3% of dog faces detected in first 100 images of human dataset.

CNN using transfer learning: The CNN model created using transfer learning with ResNet101 architecture was trained for 10 epochs.

Justification

The model created using transfer learning better performace as compared to the CNN model created from scratch.

Improvement

The model can be improved by adding more training and test data, currently the model is created using only 133 breeds of dog. Also, by performing more image augmentation, we can avoid overfitting and improve the accuracy. I have tried only with ResNet 101 architecture for feature extraction, May be the model can be improved using different architecture.

References

1. *Original repo for Project - GitHub: 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. *https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural- networks-the-eli5-way-3bd2b1164a53*
4. *http://wiki.fast.ai/index.php/Log_Loss*