Machine Learning Engineer Nanodegree

Capstone Proposal

Saurabh Agrawal

February 20th, 2018

Dog Breed Identification

Domain Background

This project is a kaggle competition taken from https://www.kaggle.com/c/dog-breed-identification.

There are too many different breeds of dogs and it is not easy to identify which breed a dog belongs to. kaggle is providing a datasets of images which has strictly dog images. There are total 120 breeds of dog covered in this training set. Although with more dog breeds training data, this solution could be extended for more than 120 breeds as well. The training set has a total of 10222 images across these 120 breeds and breed of dog in corresponding image.

The purpose of this project is to train and test the model for 120 breeds of dogs only. I expect that this model would be easy to train on other breeds of dogs and other animals like cats, horses etc but this project does not aim to test this.

Problem Statement

There are more than 360 breeds of dogs as recognised by FCI (Fédération Cynologique Internationale) as per the source https://en.wikipedia.org/wiki/List_of_dog_breeds_recognized_by_the_FCI.

It would be quite difficult for a person to identify the breed of a dog. While some of the breeds may look very similar to another one, some looks totally different. This could be a common problem if one considers the breeds of other similar animals like cats, monkeys, horses, cows, pigs etc. Even though all these different breeds have distinct features, it might be difficult for a person to remember and recognise all these features. A regular person can mostly identify the animal type (cat vs dog vs monkey) and differentiate between 4-5 popular breeds.

Datasets and Inputs

The training and test dataset is provided by kaggle. The dataset is available at https://www.kaggle.com/c/dog-breed-identification/data

There are total 4 files provided in this dataset.

- Train.zip has 10222 jpg files, each file have an image of a dog. Each image has a unique file name which acts as the id of the image. The training set covers in total 120 breeds of dog. The appendix A at the end of this document list all the breeds covered by this training data.
- Labels.csv.zip has breed label for all the training images. The format of labels.csv is “id, breed” . The id here is the name of the file from train.zip image set and breed is the breed of dog in that image.
- Test.zip has total 10357 images. We do not have the breed name of these dog images. Once we have trained the model, we will have to predict the breed of dogs in these images and upload the result to kaggle. Kaggle will tell us the accuracy of our model based on this dataset.

- Sample_submission.csv.zip is a sample file providing the template/format for the predictions on test data. Our model should generate and store the predictions in this format. This file will have image id i.e. file name in test.zip followed by the probability for all the dog breeds for the corresponding image.

Solution Statement

I intend to use the convolutional neural network for building this model. All our training data is in images and convolutional neural network works well for identifying patterns in images. For convolutional neural net implementation, I plan to use tensorflow library along with regular python libraries like numpy, sklearn, matplotlib etc. wherever applicable.

Benchmark Model

There is a test dataset provided by kaggle. The prediction for test data will not be simply the identified breed but the probability of all the breeds for an image. Based on this, kaggle will provide us the accuracy of our model. I intend to achieve a score of less than 0.3 for this and hope to get in top 50% of kaggle leaderboard for this problem.

Evaluation Metric

As this is a kaggle competition, the evaluation metric is already selected by kaggle as the “multiclass log loss” between the predicted probability and the observed target. The evaluation metric is explained at https://www.kaggle.com/c/dog-breed-identification#evaluation

Project Design

I plan to use CNN implementation for this problem. As these are images, I will check if all the images are of same resolution. If not then I will have to choose the optimal resolution and make sure to scale all the images to that level before feeding to my model.

As kaggle has decided to use log loss function for evaluation, I will also use it for train and test data and try to minimize the log loss value for the model.

For test data, I will have to store the predictions for test data in a file and then manually upload it to kaggle. Based on that kaggle will provide the score using log loss function.