Udacity Capstone Project Report
Jill Shah

Dog Breed Classifier using CNN

Project Overview:

Image classifier is one of the hot topics in the Machine Learning field. One of the most popular Udacity projects across machine learning and artificial intelligence nano degree programs. The goal is to classify images of dogs according to their breed. If the Human face is given as input it will suggest dog breed.

Python libraries like OpenCV and Tensorflow are used for all the processing. Opencv used for face detection and Tensorflow used for model building and improvement of the model.

Problem Statement:

The goal of the project is to build 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: Given an image of a dog, the algorithm will identify an estimate of the canine's breed.

Human face detector: If supplied an image of a human, the code will identify the resembling dog breed.

Datasets and Inputs:

Input Format : Image type

Dataset information: We need two types of images to test and train the model.

Dog Dataset: The dogImages folder contains 3 folders: test, train and valid. Each folder here contains 133 subfolders for dog breeds we need to classify for. There are a total of 8351 images. 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 Dataset: Human pictures are sorted by name of each human. The human dataset contains 13233 total human images which are sorted by names of humans (5750 folders). All images are of size 250x250. Images have different backgrounds and different angles. The data is not balanced because we have 1 image for some people and more than one for some.

Solution statement: First we will use the OpenCV model to check if the picture is human or not. After that we will use VGG16 model to check if the picture contains a dog. After this we will train our own CNN transfer learning model for dog breed classification.

Udacity Capstone Project Report
Jill Shah

Benchmark Model: As a benchmark model we will use a CNN model created from scratch which should have an accuracy greater than 10 percent. It is still better off than a random chance which has a probability of less than 1%.

Evaluation Metrics: As a monitoring metric we will consider accuracy. It gave us an accuracy of 68 percent.

Implementation: Below are the steps I took to build this project:

1. Download Udacity images for this project into lfw and dog images folder
2. Checked for imbalance in datasets.
3. Used OpenCV's HaarCascade classifier to identify human faces in dataset and checked for model's accuracy. Used VGG16 to identify dog pictures.
4. After this I built a CNN model from scratch and trained it and validated it. Model has 3 conv. Layers and 1 hidden layer. Pooling has been used before each layer.
5. Model when deployed on test data gives an accuracy of 10 percent and it serves as the benchmark model.
6. Finally we will build the new model using transfer learning using ResNet50. It gave us an accuracy of 68 percent.

Reflection: There are still a lot of scope to increase model accuracy with the following techniques: Image Augmentation, Increasing Dense layers and Increasing no of epochs with Dropout to decrease the chances of model overfitting. Following the above areas, I'm sure we could increase the testing accuracy of the model to above 95%.

Justification: With 68% accuracy, the model performed much better than the benchmark model.