Machine Learning Engineer Nanodegree Capstone Project
Proposal

Dog Breed Classifier using CNN

Nurbol Sakenov
August 2, 2020

Domain Background

In machine learning, image classification is accomplished by extracting key features from an image and supplying the features as inputs to the supervised model. Convolutional neural network (CNN) has gained a significant interest recently as it provides excellent results in image classification. In this project, a CNN model will be used to classify a dog breed.

Example dog breed output.

Problem Statement

The goal of the project is to create a CNN machine learning model to classify a dog breed given a real world image of a dog. In addition, if the model is supplied with an image of a human, the model should provide a resembling dog breed.

Dataset and Inputs

The project dataset is provided by Udacity:

- dog images dataset: 8351 images in total:
    - 133 total dog categories.
    - 6680 training dog images.
    - 835 validation dog images.
    - 836 test dog images.
- human images dataset: 13233 images in total

Solution Statement

The solution involves three steps:

1. For human image detection, OpenCV's implementation of Haar feature based cascade classifiers will be used.
2. In order to detect dog-images, pretrained VGG16 model will be used.
3. After the steps one and two are completed, the image will be supplied to the CNN model which will assign the best matching dog breed out of 133 dog breeds.

Benchmark Model

The benchmark model will be a simple CNN classifier created from scratch, which is expected to have accuracy of at least 10%. Then the model will be improved by learning on the train dataset and using transfer learning.

- After testing, the trained CNN model should have accuracy of at least 60% and above.

Evaluation Metrics

The evaluation metric will be accuracy, which is calculated by comparing predictions to the true label and dividing the count of total correct results by the total number of predictions. The formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where:

TP - true positive
TN - true negative
FP - false positive
FN - false negative

Project Design

Python libraries to be used*: Numpy, CV2, tqdm, torch, torch.vision
* not exhaustive, requirements.txt will be provided in the project repository.

Workflow:
Step 0:
Loading training/test dataset for human and dog images and getting understanding of dataset sizes. Loading a sample image for check.

Step 1:
Detecting human faces using the OpenCV's implementation of Haar feature-based cascade classifiers.

Step 2:
Detecting dog images using a pre-trained VGG-16 model.

Step 3:
Creating a CNN to Classify Dog Breeds from scratch. The classifier should reach accuracy of at least 10%. Additional steps such as resizing, rotation, and cropping will be included.

Step 4:
Building a CNN model to classify dog breeds using transfer learning. The model will be trained in *n* epochs, a loss function, and optimizer will be specified. The accuracy of the model should be greater than 60%.

Step 5:
Writing an algorithm that outputs a dog breed as the prediction if a dog is detected. In case a human is detected, the algorithm will return the resembling dog breed.

Step 6:
Testing the accuracy of the model with images from the dataset.

References

1. GitHub repository of the original project:
https://github.com/udacity/deep-learning-v2-pytorch/tree/master/project-dog-classification

2. Datasets:
https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/dogImages.zip
http://vis-www.cs.umass.edu/lfw/lfw.tgz