

Machine Learning Engineer Nanodegree Capstone Proposal

DOG BREED CLASSIFIER

Oluwatola Oluwafemi Tofade

1. Domain Background

Dog breed classification is a popular and challenging problem in the Machine Learning Space. Dogs generally look similar but there exist certain features that make it a member of a particular dog breed type. So also, dogs possess varieties of colors, sometimes a particular breed type is identified by its colors. This becomes a complex computer vision task for humans as one would need to remember all the features of a particular breed and also must be able to distinguish these features from other breeds. One solution to this complexity would be to combine various models where each model would be specific such as identifying humans, identifying dog breeds and identifying humans faces that looks like a dog breed.

2. Problem Statement

The goal of this project is to build a pipeline to process real-world user-supplied images. The algorithm predicts the breed of a dog given an image and if a human image is passed to it, it predicts the closest matching dog breed. If Neither, it should return that it can't detect if the image is human or a dog.

3. Datasets and Inputs

The input format for this project must be images. This images/dataset were provided by Udacity for training, validating, and testing of the CNN models for the different tasks.

- **Dogs Dataset**

The dog image dataset has 8,351 images which were sorted into training (6,680 Images), testing (836 Images) and validating (835 Images) directories. Each of these directories had 133 folders corresponding to dog breeds. Each image was unique from each other and varied in size from each breed.

- **Humans Faces Dataset**

The humans image dataset has 13,233 images which were stored in 5,750 folders corresponding to people names. The images were all unique from each other.

4. Solution Statement

Convolutional Neural Network (CNN) model can be used to solve this complex multi-classification problem.

What is CNN in simple words?

A convolutional neural network (CNN) is a type of artificial neural network used in image recognition and processing that is specifically designed to process pixel data.

The solution consists of three steps, which are:

- Detection of humans – Here, this can be fulfilled using OpenCV's implementation of Haar feature-based cascade classifiers
- Detection of dogs: Pre-trained VGG-16 model will be used for to detect dog images.
- Detection of the Breed: Once the image is identified as dog/human, it will passed to the CNN to predict dog breed for dog image and resembling dog breed for human image.

5. Benchmark Model

- The CNN model created from scratch must have accuracy of at least 10% on test data, which would be better than a random guess which has an accuracy of 1/133% (less than 1 %). This is the baseline accuracy.
- The CNN model created using transfer learning must have an accuracy of at least 60% on the test data.

6. Evaluation Metrics

For this multi-classification problem, 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 of prediction based on how much it varies from actual label and this will help in evaluating the model.

7. Project Design

Here is how I intend to go about this project:

- I. Import the necessary dataset and libraries, Pre-process the data and create train, test and validation dataset. Perform Image augmentation on training data.
- II. Detect human faces in images using OpenCV's implementation of Haar feature based cascade classifiers.

- Convert image into grayscale for input
 - Face detector function to detect face and returns true or false
- III. Detect dogs in images using pretrained VGG16 model.
 - IV. Create a CNN model to classify dog breeds from scratch, train, validate and test the model.
 - V. Create a CNN to Classify Dog Breeds using Transfer Learning with resnet50 architecture. Train and test the model.
 - VI. Write an algorithm to accept an image file path as input and detect if image contains dog or human. Predict breed for dog image and resembling dog breed for human image and output error if neither dog nor human is detected
 - VII. Test the algorithm with different images to see if model provides correct prediction or not.

8. References

- CNN Model: https://en.wikipedia.org/wiki/Convolutional_neural_network
- Udacity Dogs Data set: [Dogs Dataset](#)
- Udacity Human Data set: [Humans Dataset](#)
- PyTorch Models - <https://pytorch.org/docs/stable/torchvision/models.html>
- <https://github.com/KwokHing/udacity-project-dog-classification>
- <https://github.com/Jayshree-S/dog-breed-classifier-CNN>