

Dog Breed Classifier using CNN

Domain Background:

Image Classification: Image classification/recognition is a much studied problem in Machine Learning. This model will use multi-class classification using supervised learning.

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.

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 and F1 score of the model. F1 considers both recall and precision which makes it better for comparison across different models.

Udacity Capstone Project Proposal

Jill Shah

Project Design: Below are the steps I am looking to follow to build this project:

1. Download Udacity images for this project.
2. Data exploration to check for types of images, number of images and check for imbalance in datasets.
3. Use OpenCV's HaarCascade classifier to identify human faces in dataset and check for model's accuracy.
4. Explore other models for above task.
5. Use VGG16 to identify dog pictures.
6. All the above steps will require some pre-processing of the images.
7. After this we will build a CNN model from scratch and train it and validate it.
8. We will deploy the model on our test data to see if the model gives an accuracy of atleast 10 percent.
9. If it does, it will serve as our bench mark model.
10. Finally we will build the new model using transfer learning. Hopefully that will give us an accuracy greater than 60 percent.