# Dog Breed Classifier

## Domain Background

Convolutional Neural Networks are a type of NNs which uses much less number of parameters to produce good results for computer vision tasks like image classification, segmentation, object detection, etc. CNNs are normally quite deep and consist of different types of hidden layers like convolutional layer, pooling layer,  batchnorm layer and fully connected layers as well. A lot of success in applying deep learning in computer vision tasks comes from using CNNs. But we also do not train a model from scratch for a new application. We have pretrained models trained on huge datasets like Imagenet, easily accessible through Pytorch. This reduces the training time for a new model trained on a different dataset as the model weights doesn't start from random and the model is able to determine shapes and figures easily.

## Problem Statement

In this project, we will try to classify dogs as per their breeds. This is a supervised learning problem as we will be using the images and its labels to train a model. At the end of this project, the code should accept any user-supplied image as input. If a dog is detected in the image, it will provide an estimate of the dog's breed. If a human is detected, it will provide an estimate of the dog breed that is most resembling. The image below displays potential sample output of the finished project:

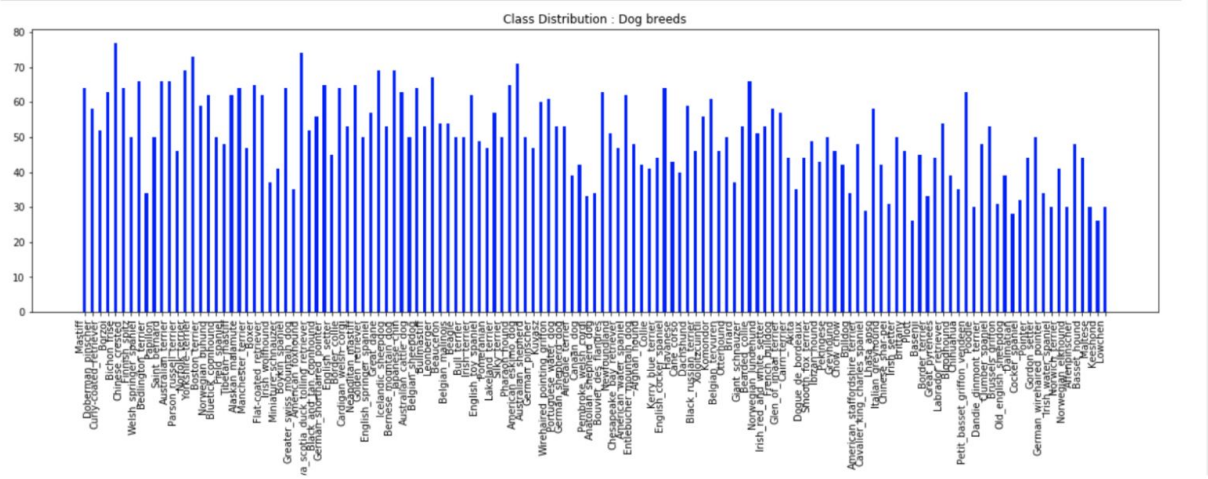

## Datasets and Inputs

There are two datasets used for this project which are provided by Udacity:

- dog dataset : The dog dataset consists of 8,351 dog images across 133 breeds. The This dataset would be divided between training, validation and test set. The dataset will be distributed as follows:
    - Number of training images:  6680

- Number of validation images:  835
- Number of test images:  836

The class distribution of the training set is not evenly balanced as shown below:




- [human dataset] : The human dataset consists of 13,233 human images.

## Solution Statement

The project will use CNN architecture to solve this problem. We will use a pretrained model and further train it on the dog dataset to classify a given image of a dog as per it's breed.

## Benchmark Model

A CNN architecture similar to VGG16 would be used be trained from scratch and it would be used as benchmark for this project.

## Evaluation Metrics

The model will be evaluated on the basis of the accuracy it achieves on the test dataset.

## Project Design

1. Import Datasets
2. We will use OpenCV's implementation of [Haar feature-based cascade classifiers] to detect human faces in images.

3. We use a pre-trained model to detect dogs in images.
4. Preprocessing: We would resize and normalize the input images. We would also use some transforms like flip and rotate to augment the dataset.
5. Create and train a CNN a model to classify dog breeds.
6. Use a pretrained model and finetune it on the dog training set to classify dog breeds.
7. Write an algorithm that accepts a file path to an image and first determines whether the image contains a human, dog, or neither.
    - If a **dog** is detected in the image, return the predicted breed.
    - If a **human** is detected in the image, return the resembling dog breed.
    - if **neither** is detected in the image, provide output that indicates an error.
8. Test the Algorithm on sample images.