# Udacity Machine Learning Engineer Nanodegree 2020

## Capstone Proposal

# Dog Breed Classifier Using CNN

**Saurabh Dubey**

**16th April 2020**

### 1. Domain Background

Dogs are one of the most common domestic animals you will find at most of the homes. As the number of dogs is increasing, a lot of issues associated with dogs are also increases. Issues such as Rabies, vaccination control, etc. So to control such issues we want to identify each dog breed so that we can take appropriate measures to solve this issue. So the idea here is to build the model pipeline which can be trained on a lot of different dogs breed images and then identify or classify the given real-world image of a dog in different breed types. This is a label type of problems and it has multi-class output so it's a multi-class classification problem so we will use a supervised learning approach to solve this problem.

I want to deploy this project model on the web app so that it can take real-world images of dogs and be able to give predictions. By deploying it can be used for a lot of animals shops to classify and make sure that some different breeds of dogs to be put in different cell.

### 2. Problem Statement

The main objective of this project will be to use Deep Learning Convolutional Neural Network with Transfer Learning to classify the dogs breed.

Given an image of a dog, we want to be able to identify an estimate of the canine's breed with a corresponding likelihood score.  If supplied with an image of a human, the code will identify the resembling dog breed.

### 3. Datasets and Inputs

The dataset for this project was provided by Udacity. The dataset has pictures of Dogs and Humans.

The dataset is in Image format. The dog dataset contains train, valid and test folders and each folder contains every 133 samples of different dogs canine's breed. The number of images in each folder is not the same i.e some dogs breed folder, the number of images is less and in some, the number of images is more. Pixel size also varies for a lot of images.

The human dataset contains 5749 items of humans. The human dataset, when supplied to our model will give resembling dog breed. Each folder has 1 image of input. Each image of size 250x250.

## 4. Solution Statement

The dataset is in Image format, so we will Convolutional Neural Network to solve this problem. CNN works best when the input is in Image format. This is a Multiclass Classification Problem i.e. we will classify the given image of a dog in one of the 133 breeds.

The 1st Step in this solution is that it will detect the humans in the given image. We will use the OpenCV implementation of Haar feature-based cascade classifiers to detect human faces in images. OpenCV provides many pre-trained face detectors

The 2nd Step is that we will detect dogs in a given image. In this, we will use a pre-trained model to detect dogs in a given image. We will use the VGG16 pre-trained model

The 3rd Step is that we will create a CNN which will classify the dogs breed out of 133 breeds.

## 5. Benchmark Model

The CNN model created from scratch must have an accuracy of atleast 10%. This can confirm that the model is working beacuase a random guess will provide a correct answer roughly 1 out of 133 times, which corresponds to an accuracy of less than 1%.

The CNN model created using Transfer Learning must have an accuracy of 60% and above.

## 6.Evaluation Metrics

As this is a Multiclass Classification Problem, the evaluation metrics used for this type of problem is Multi class log loss. The nature of our dataset is also imbalanced so the accuracy is not a good measure for evaluating the model.

The Log loss finds the difference between the actual label versus the predicted label and this helps us to find the error of the model, so that we can improve it further. So, Log Loss is a goo measure for evaluating the our Multiclass classification problem model.

## 7. Project Design

Step 1: Import Datasets of dogs and humans.

Step 2: Used the pre-trained OpenCV implementation of Haar feature-based cascade classifiers to detect human faces in images

Step 3: Use pre-trained VGG16  model to detect dogs in a given image.

Step 4: Create a CNN from sratch to classify dogs breed

Step 5: Create a CNN with the helped of pre-trained Transfer Learning with resnet101 to classify dogs breed.

Step 6: Write an Algorithm that will combine dog detector and human detector

If a dog is detected in the image, return the predicted label

If a human is detected in the image, return the resembling dog breed

If neither dog nor human is detected in the image, provide output that indicates an error

## References

**1.** Original repository for project: - https://github.com/udacity/deep-learning-v2-pytorch/tree/master/project-dog-classification

**2.** https://github.com/pytorch/vision/blob/master/torchvision/models/resnet.py