MACHINE LEARNING NANODEGREE
CAPESTONE PROJECT PROPOSAL: DOG BREED CLASSIFICATION

BACKGROUND:

This project deals with classifying dog breeds given images as input. Differentiating dog breeds is a challenging problem because there are low inter-breed and high intra-breed variation; to describe it more clearly, there are relatively few differences between breeds and relatively large differences within breeds, differing in size, shape, and color. Dogs are both the most morphologically and genetically diverse species on Earth. This problem is not only challenging but also the method used for solving this problem will apply to other fine-grained image classification problems [2]. For example, to identify breeds of cats and horses as well as species of birds and plants or even models of cars. Convolutional neural networks (CNNs) have been mostly used in applications such as object classification, scene recognition, and many other applications [1][3]. This capstone project aims to apply deep learning techniques for classifying dog breeds.

PROBLEM STATEMENT:

The project aims to design a CNN for identifying the breed of the dogs. At the end of this project, the code will 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.

DATASETS AND INPUTS:

There are two different sets of data to work with:

1. Dogs Dataset: It contains 133 folders. Each folder corresponds to a different dog breed. There are 8351 dog images in the dataset.
2. Human Dataset: It contains 13233 human images.

The datasets are provided in Udacity's workspace.

SOLUTION STATEMENT:

The problem which we are dealing with is a multi-classification problem. As CNN provides better accuracy for the multi-classification model. I will be designing a CNN model for identifying the breed of the dogs. Also designing a CNN using transfer learning techniques will help in attaining

high accuracy and it will take less training time as we'll be using a pre-trained model for extracting features.

BENCHMARK MODEL:

The classes in the dataset are not balanced. It is very challenging to attain a high accuracy for this dataset. So, designing an efficient CNN model from scratch for classifying the dog breed will help in attaining good accuracy.

EVALUATION METRICS:

- As the classes of the dataset are not balanced, cross-entropy loss also known as log loss can be used for handling the imbalance.
- The performance of the model can be evaluated by accuracy metrics.

  Accuracy = number of correct predictions / total number of predictions

PROJECT DESIGN:

Step 1: Importing Datasets

Step 2: Data Preprocessing

For data preprocessing I will resize the images before feeding them to the network as the dataset contains images of various sizes. Because CNN structure presumes that the input would have a certain shape, like 224 x 224 x 3. I'm also planning to include data augmentation techniques like flipping and rotation because data augmentation is an easy way to extend a dataset and improve the model's performance.

Step 3: Detecting Humans

Step 4: Detecting Dogs

Step 5: Designing a CNN to Classify Dog Breeds (from Scratch)

As the project requirements are to only achieve 10% or greater test accuracy. I'm planning to take the simplest model from the VGG16 paper [4] and further simplified it.

Step 6: Designing a CNN to Classify Dog Breeds (using Transfer Learning)

As the VGG16 network was also trained on dogs, among other things, it contains useful high-level feature information in the later convolutional layers. So, I'll use the entire feature extractor part

from the VGG16 pretrained model keeping the weights constant, to avoid overfitting when training on a small new dataset like the dataset we have. Then I'll replace the classifier part with my dog breed classifier.

Step 7: Writing an algorithm that accepts a file path to an image and first determines whether the image contains a human, dog, or neither. Then,

- 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 an output that indicates an error.

Step 8: Testing the algorithm on sample images

REFERENCES:

1. A. Krizhevsky, I. Sutskever, and G. Hinton. "Imagenet classification with deep convolutional neural networks", Advances in neural information processing systems. 2012.
2. J. Liu, A. Kanazawa, D. Jacobs, and P. Belhumeur. "Dog Breed Classification Using Part Localization", Computer Vision–ECCV 2012. Springer Berlin Heidelberg, 2012. 172-185.
3. Hsu, David. "Using Convolutional Neural Networks to Classify Dog Breeds", Stanford University.
4. K. Simonyan, A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". ICLR 2015.