Machine Learning Engineering Nanodegree

Capstone Proposal

Gowtham Mallikarjuna

Abstract

This project is about the classification of images to different dog breed groups. Given the image of dog the machine learning model will classify to appropriate dog breed. If the human image is given it will be classified in to the closest resembling dog breed.

Domain Background

This project is particularly interesting for me because I can make use of all the good stuff from the latest advancement in computer vision field. I will be using deep learning with CNN (convolution neural networks) to identify the features to classify each test images into corresponding dog breed classification.

I had done similar work previously to classify cats and dogs using keras (https://github.com/gowtham91m/cats-and-dogs-classification). It was binary classification model using CNN with keras. This problem to identify dog breed and similarity of human face to one of 133 provided dog breeds.
I will be implementing the deep learning using pytorch. Although keras gives very high level easy abstraction of neural network implementation, pytorch provides some great advantages with dynamic graphs, better debugging capabilities and flexibility to play and fine tune the network parameters.
Will also be using pre trained network using transfer learning technique that would identify the basic image features instead of training the entire network from scratch.

Problem Statement

Build a deep learning model that can be used within an app or website to process the images. Given an image of dog algorithm will estimate the breed. Given the image of human algorithm will identify the resembling dog.

Datasets and Inputs

Data source:
dogimages_url - https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/dogImages.zip
humanimages_url - https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/lfw.zip

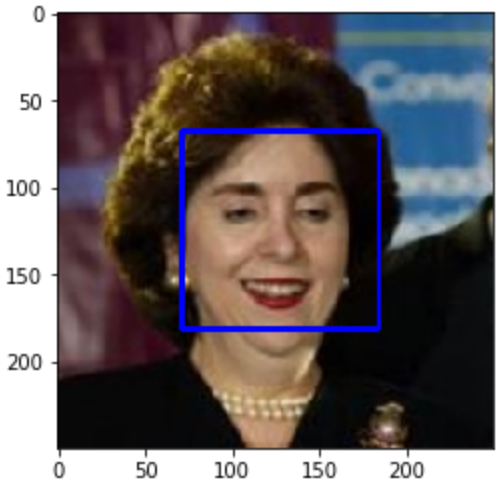
1. Dog dataset - images separated in 133 folders each corresponding to each different dog breed. There are 8351 total dog images.
2. Human dataset. There are 13233 total human images.

Dog dataset is used for training the model. The data is already split in to train, validation and test seperated in respective folders.
Within each of the train, validation and test folders the images are segregated in 133 folders for each of the dog breed classification.
Images in train folder will be used for training.
Model will be validated using images in valid folder.
Finally the model performance will be tested using the images in test folder.

Will be using keras/ pytorch data loader features to load the data from the directory as is.

The data looks fairly balanced looking at the count of images in each of the directory (as shown below in the next section)

Some example images are given below



Solution Statement

Algorithm will locate the face of the human/dog in the given image and identify the dog breed based on the identified features using Convolution Neural Networks.

Data will be normalized before feeding to the training algorithm. Also some data augmentation will be performed to make the model more robust. For example random rotation and scaling, Since a dog will be a dog in which every the directory or the aspect ratio the photo is taken. Our algorithm should be able to make the right prediction.

Solution will be implemented in 3 steps.

- In the first step the input image is predicted if it is a dog image or a human face image.
- Second step will detect the dog classification among 133 classes using transfer learning with the pre trained weights for initial part of the network.
- Third step is decision algorithm will invoke the function implemented in first step to identify if it is a dog or human and call the second function to predict the dog breed to print the output.

Benchmark Model

My Benchmark model will be initial CNN model built from scratch with the accuracy of about 12%.
I will try to improve the model further using transfer learning.

Evaluation Metrics

Here is the distribution of count of number of images in each of the dog classification.

64 : ./001.Affenpinscher
58 : ./002.Afghan_hound
52 : ./003.Airedale_terrier
63 : ./004.Akita
77 : ./005.Alaskan_malamute
64 : ./006.American_eskimo_dog
50 : ./007.American_foxhound
66 : ./008.American_staffordshire_terrier
34 : ./009.American_water_spaniel
50 : ./010.Anatolian_shepherd_dog
66 : ./011.Australian_cattle_dog
66 : ./012.Australian_shepherd
46 : ./013.Australian_terrier
69 : ./014.Basenji
73 : ./015.Basset_hound
59 : ./016.Beagle
62 : ./017.Bearded_collie
50 : ./018.Beauceron
48 : ./019.Bedlington_terrier
62 : ./020.Belgian_malinois
64 : ./021.Belgian_sheepdog
47 : ./022.Belgian_tervuren
65 : ./023.Bernese_mountain_dog
62 : ./024.Bichon_frise
37 : ./025.Black_and_tan_coonhound
41 : ./026.Black_russian_terrier
64 : ./027.Bloodhound
35 : ./028.Bluetick_coonhound
74 : ./029.Border_collie
52 : ./030.Border_terrier
56 : ./031.Borzoi
65 : ./032.Boston_terrier

45 : ./033.Bouvier_des_flandres
64 : ./034.Boxer
53 : ./035.Boykin_spaniel
65 : ./036.Briard
50 : ./037.Brittany
57 : ./038.Brussels_griffon
69 : ./039.Bull_terrier
53 : ./040.Bulldog
69 : ./041.Bullmastiff
63 : ./042.Cairn_terrier
50 : ./043.Canaan_dog
64 : ./044.Cane_corso
53 : ./045.Cardigan_welsh_corgi
67 : ./046.Cavalier_king_charles_spaniel
54 : ./047.Chesapeake_bay_retriever
54 : ./048.Chihuahua
50 : ./049.Chinese_crested
50 : ./050.Chinese_shar-pei
62 : ./051.Chow_chow
49 : ./052.Clumber_spaniel
47 : ./053.Cocker_spaniel
57 : ./054.Collie
50 : ./055.Curly-coated_retriever
65 : ./056.Dachshund
71 : ./057.Dalmatian
50 : ./058.Dandie_dinmont_terrier
47 : ./059.Doberman_pinscher
60 : ./060.Dogue_de_bordeaux
61 : ./061.English_cocker_spaniel
53 : ./062.English_setter
53 : ./063.English_springer_spaniel
39 : ./064.English_toy_spaniel
42 : ./065.Entlebucher_mountain_dog
33 : ./066.Field_spaniel
34 : ./067.Finnish_spitz
63 : ./068.Flat-coated_retriever
51 : ./069.French_bulldog
47 : ./070.German_pinscher
62 : ./071.German_shepherd_dog
48 : ./072.German_shorthaired_pointer
42 : ./073.German_wirehaired_pointer
41 : ./074.Giant_schnauzer
44 : ./075.Glen_of_imaal_terrier

64 : ./076.Golden_retriever
43 : ./077.Gordon_setter
40 : ./078.Great_dane
59 : ./079.Great_pyrenees
46 : ./080.Greater_swiss_mountain_dog
56 : ./081.Greyhound
61 : ./082.Havanese
46 : ./083.Ibizan_hound
50 : ./084.Icelandic_sheepdog
37 : ./085.Irish_red_and_white_setter
53 : ./086.Irish_setter
66 : ./087.Irish_terrier
51 : ./088.Irish_water_spaniel
53 : ./089.Irish_wolfhound
58 : ./090.Italian_greyhound
57 : ./091.Japanese_chin
44 : ./092.Keeshond
35 : ./093.Kerry_blue_terrier
44 : ./094.Komondor
49 : ./095.Kuvasz
43 : ./096.Labrador_retriever
50 : ./097.Lakeland_terrier
46 : ./098.Leonberger
42 : ./099.Lhasa_apso
34 : ./100.Lowchen
48 : ./101.Maltese
29 : ./102.Manchester_terrier
58 : ./103.Mastiff
42 : ./104.Miniature_schnauzer
31 : ./105.Neapolitan_mastiff
50 : ./106.Newfoundland
46 : ./107.Norfolk_terrier
26 : ./108.Norwegian_buhund
45 : ./109.Norwegian_elkhound
33 : ./110.Norwegian_lundehund
44 : ./111.Norwich_terrier
54 : ./112.Nova_scotia_duck_tolling_retriever
39 : ./113.Old_english_sheepdog
35 : ./114.Otterhound
63 : ./115.Papillon
30 : ./116.Parson_russell_terrier
48 : ./117.Pekingese
53 : ./118.Pembroke_welsh_corgi

31 : ./119.Petit_basset_griffon_vendeen
39 : ./120.Pharaoh_hound
28 : ./121.Plott
32 : ./122.Pointer
44 : ./123.Pomeranian
50 : ./124.Poodle
34 : ./125.Portuguese_water_dog
30 : ./126.Saint_bernard
41 : ./127.Silky_terrier
30 : ./128.Smooth_fox_terrier
48 : ./129.Tibetan_mastiff
44 : ./130.Welsh_springer_spaniel
30 : ./131.Wirehaired_pointing_griffon
26 : ./132.Xoloitzcuintli
30 : ./133.Yorkshire_terrier

We can see the count of images in each of the classes is fairly normally distributed for a multiclass classification problem with 133 classes.
I will be using accuracy as my evaluation metric.

Project Design

Download dataset
Resize and normalize the dataset.
Perform data augmentation to make the model more generalized.
Build an algorithm to predict if the given image has human face
Build an algorithm to predict if the given image contains dog
Build a classifier to classify the image into one of the 133 dog breeds
Given an image
    if the image has a human face, algorithm will predict the closest resemblance dog breed
    If the image has a dog, algorithm will predict the dog breed
    If the image doesn’t contain both, it will throw an error