Final report of

Dog breed classification capstone project

Ali Nasser Mohamed

March 25, 2021

Project overview :

The problem of dog breed classification is a famous problem in machine learning and deep learning .the problem is about predicting the breed of a dog after from it is image if we were given an image of a dog and predicting the resembling dog breed if we were given a picture of human .the project is multi-class classification project in which we are going to use supervised learning .in the project I am going to use CNN (conventional neural network) to solve it .I choose this project because I want to go deeper in deep learning and neural network

Project goal:

My goal in this project is to build a machine learning model that can predict dogs breed and assembling dog breed to human face

The algorithm here will do two tasks

1-Human face detector: if we give the model an image of human it should predict the resembling dog breed of the human

To achieve this task we will built face detector that use OpenCV's implementation using Haar feature based cascade classifier that will return true if the image has human face and false otherwise
and if it returns true we will feed the image into CNN model that classify it to the resembling dog breed

2-dog breed detector: if we give the model an image of dog it should predict the canines breed of this dog

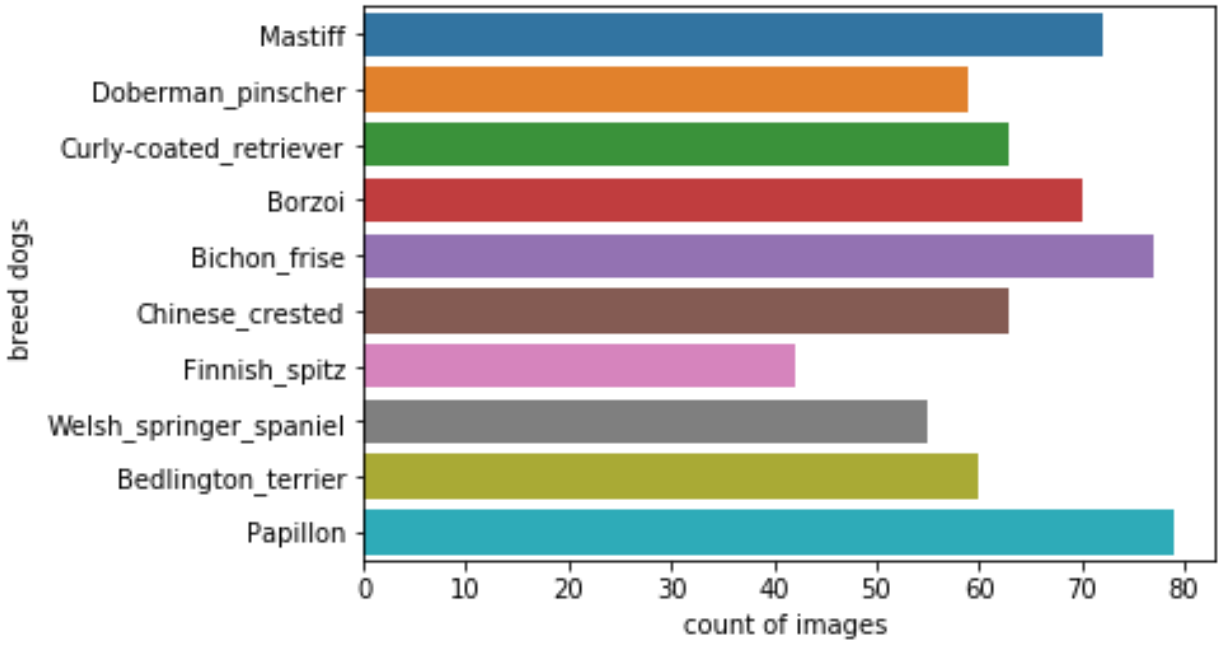
To achieve this task we will built CNN model(from scratch and using transfer learning) that takes in the image of the dog and return it's breed

Data description:

The data set used in this project is provided by Udacity. the data set contains images of dogs and humans so the input type to the model is image type .the data is divided into two categories :

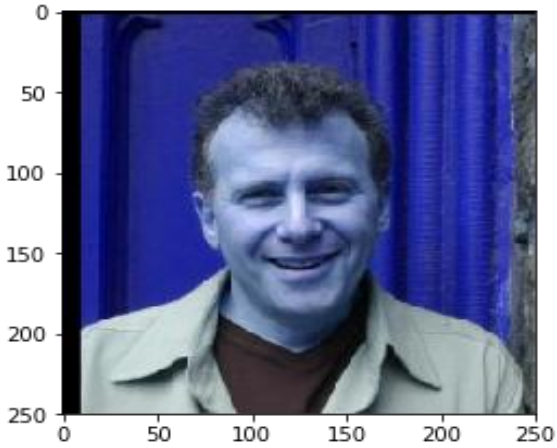
1-dog images data set: the data set contains 8351 total dog images with different sizes and backgrounds

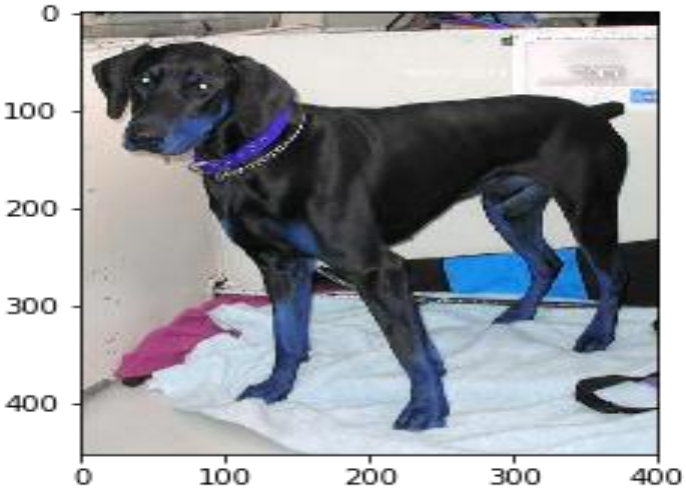
The dog image data set is divided into train (6680 images), validation (835 images) and test (836 images) directories and each of these directories have 133 folders corresponding to 133 dog breeds the data is not balanced because not all the folders contain the same umber of images as we see in the distribution of ten dog breeds below



2-human images dataset: the data set contains 13233 total human images with different sizes and backgrounds

Some image samples from the datasets:





Some statistics about dogs data set :

The mean value of images in each dog breed is 62 with standard deviation of 14.7

tha maximum value of image in all dog breeds is 96 and the minimum value is 33

Algorithms and Teqniques:

To solve this problem we are going to use CNN(conventional neural network),CNN is a deep learning algorithm used to classify images by assigning a new values to it's weights and bias .the solution has three steps:

firstly, we are going to use OpenCV's implementation of Haar feature-based cascade classifiers to detect human faces in images and built human face detection function with it that accept image and returns true if the image is human and false otherwise

Secondly, we are going to use VGG16 model to detect dogs in images

Vgg16 is very suitable for this process because it is built on IMGNET data set which is similar to the data set we use in our project

We will build dog detection function with it that accept image and returns true if the image is dog and false otherwise

Finally, we are going to pass the image to CNN to predict the breed that matches the best out of 133 breeds

-firstly we build CNN model from scratch (with random weights and bias) then we train it on the training data set and test it with test dataset

-secondly , we are using transfer learning model (VGG16) to improve the CNN model from scratch we built before

The transfer learning model has advantages of pretrained model with modified weights and bias that make it time saving and help in getting high prediction accuracy

Metrics:

The data is split into training, validation and testing datasets, the model is trained using the raining dataset ,we use testing data for predict the performance of the model on unseen data ,we will use accuracy for predicting model performance on test data set

Accuracy=Number of correctly classifies items/total number of items

Also during training, we compare test data with validation dataset and calculate multi class log loss function to find best performing model as log loss takes into account the amount of uncertainty of prediction and how many it varies from actual labels and help us evaluating the model

Benchmark model:

We can see that the CNN model is working if we get accuracy greater than 10% as the accuracy of random guess in classification of 133 breed is less than 1%(exactly one per every 133 times)

Data processing:

firstly, we resize image to 256 and crop it from center to size of (224*224) i used this size because it most suitable for my model

To overcome overfitting, I used random horizontal flip and random rotation with 30 degree and this decreases overfitting in the data

Implementation:

Firstly I build CNN model from scratch as follows

I defined 6 convolution layers ,six fully connected layers ,and drop out layer(with p of 0.05) I had experimented many values of p (.25 , 0.1 ,0.05) and the best value was 0.05 I passed the data through the convolution layer and after each convolution layer I do max pooling of (2 * 2) then passed the data through fully

connected layers with RELU activation function and the last function has output of 133 classes

The accuracy produced from this model with 11% and this was indication that the model works good as the accuracy of random guess in classification of 133 breed is less than 1%(exactly one per every 133 times)

Refinement:

When using CNN model from scratch we get accuracy of about 14% so I decided using transfer learning to refine the model as follows

firstly, I experimented using resnet101 architecture but it was slow and can not get the required accuracy from it

so I decided using VGG16 model and it works good

the VGG16 model is suitable for this task as it built on IMFNET data set

I built the last classification layer with input size of 25088 and output size of 133 classes which belong to each dog breed

the last accuracy i get was 83%

Model Evaluation and Validation:

Human face detector the human face detector was created using OpenCV's implementation using Haar feature based cascade classifier

98% of human faces was detected in first 100 images of human face data set and 17% human faces was detected in first 100 images of dog data set

Dog face detector :the dog face detector was created using VGG16

100 % of detected dogs was detected in first 100 images of dogs data set and about 2% detected dogs in first images of human data set

CNN model was created using transfer learning of VGG16 and get accuracy of 83%

Justification:

the output of the CNN model from scratch was good and beats the benchmark model and can get accuracy of 11%

the output was better than I expected as I get accuracy of 83% when training the data with 10 epochs

Improvements of the model :

1- do some extra image augmentation

2- training the model for more than 10 epoches

3-do some hyper parameter tunning