Dog Breed Classifier using CNN

Domain Background:

There are a set of famous problem statements in the field of Artificial intelligence, one of which is the classification of a dog breed. This project basically functions as a dog breed classifier, wherein when given a dog's image it identifies what breed it is and if a human's image is given it tries checking if the face resembles any dog breed and mention it. For this I will create a CNN to classify the dog breeds and then do multi-class classification through supervised machine learning. Later I will build a sequence to process the images that a user sends and classify the breed of the dog if it is a image of a dog. I was interested in this particular project since this is really a fun project to work on and moreover it gives me the opportunity to build and deploy cool machine learning models, and hence I have chosen this as my capstone project.

One of the main advantages of using a CNN is that it has a good accuracy in image classification problems. The CNNs represent a huge breakthrough in image recognition. They're most commonly used to analyse visual imagery and are frequently working behind the scenes in image classification. Image classification is the process of taking an image as an input and sending an output which specifies a class (for example a 'cat' or a 'dog') or an output which is a probability which specifies that the input belongs to a particular class (for example 'there's a 90% probability that this input is a cat')

Different people use different types of CNN. I have taken use of the ResNet (residual network) as they are easy to optimize and they can easily gain accuracy from greatly increased depth, producing results which are better than previous networks. There are other people who use AlexNet as an option.

Problem Statement:

The problem statement is that the model that I build should be able to process the images that is given by an user and be able to identify the breed of a dog if an image of a dog is given and if an image of a human is given it should see if the human's face resembles any one breed of a dog and should mention it.

Datasets and Inputs:

For the training of my model I will be using images as the input since the model should identify the breed of a dog. I will be using the dataset provided by Udacity. The dataset will contain both dog images and human images.

The dataset is classified into types:

DATASET-1: The dataset-1 totally has 8351 images of dogs, they are divided into 3 directories. Directory 1 is the 'train' which has 6,680 Images, Directory 2 is the 'test' which has 836 Images and Directory 3 is the 'valid' which has 835 Images. There are 133 folders similar to dog breeds in each of this directory. The images may have a lot of dissimilarities, either they may not have the same image size or there may be a difference in the background or the surrounding environment. The images provided for each breed may vary, one breed may have 1 image while another breed may have 10 images and hence the dataset is not balanced.

DATASET-2: The dataset-2 totally has 13233 images of humans, they are divided by the names of the human being and it has 5750 folders in it. A lot of dissimilarities can be observed in the human images, different background or environment. However the size of the image is the same (250 x 250). Just like the 1st dataset the dataset here is not balanced as there maybe 1 image for a person while there might be 5 images for others.

Solution Statement:

I implemented convolutional neural network(CNN) since we perform multiclass classification to identify the breed of a dog. Basically a CNN is a class of deep neural networks, most commonly used to look over a visual image. The CNN takes in an image as an input data, and look into almost all the aspects of the image and be able to differentiate one image from another.

My model first imports the datasets. It then tries to detect the human faces if any in the given images by using OpenCV algorithm. After this the model tries to detect dogs in images by using a pretrained VGG16 model. The model passes the image to the CNN after it identifies whether the given input is of a dog or of a human. The CNN then processes the image. If the image is of a dog it predicts the breed that matches the best out of the 133 breeds. If the image of a human is given it should see if the human's face resembles any one breed of a dog and mention it.

Benchmark Model:

A benchmark model is basically a simple or historical model or a result to compare the defined solution to.

For this purpose I will create a CNN model from scratch and this model should attain a test accuracy of a minimum 10%. The 10% is kept as a threshold so as to confirm that the model is working because a random guess will provide a correct answer roughly 1 in 133 times, which corresponds to an accuracy of less than 1%.

I will then create a CNN model using “transfer learning” and this model must have an accuracy more than 60%.

Evaluation Metrics:

We use 2 evaluation metrics “log loss” and “accuracy”. The log loss measures the performance of a classification model where the prediction input is a probability value between 0 and 1. Accuracy is easily suited for a multiclass classification problem. Accuracy is the proportion of true results among the total number of cases examined.

Accuracy formula: (TP+TN)/(TP+FP+FN+TN)

Project Design:

Step 1: The model first imports the dataset and libraries, it then pre-processes the data and creates 3 datasets namely ‘train’, ‘test’ and ‘validation’. Image augmentation is then performed on the training data.
Step 2: The model first tries to detect the human faces in the image using OpenCV algorithm.
Step 3: The model then tries to detect the dog images using pretrained VGG16 model.
Step 4: A CNN is created from scratch to classify dog breeds. It trains, validates and tests the model.
Step 5: A CNN is created using Transfer Learning with resnet101 architecture to classify dog breeds. It trains, validates and tests the model.

Step 6: An algorithm is written such that if a dog's image is given the model should identify what breed it is from and if an image of a human is given the model should try checking if the face resembles any dog breed and mention it. If neither is detected, provide output that indicates the error

References:

1. https://github.com/udacity/deep-learning-v2-pytorch/tree/master/project-dog-classification
2. https://towardsdatascience.com/the-5-classification-evaluation-metrics-you-must-know-aa97784ff226#:~:text=A.&text=Accuracy%20is%20the%20quintessential%20classification,as%20a%20multiclass%20classification%20problem.&text=Accuracy%20is%20the%20proportion%20of,total%20number%20of%20cases%20examined.
3. https://www.learnopencv.com/tag/image-classification/
4. https://en.wikipedia.org/wiki/Convolutional_neural_network
5. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53
6. http://wiki.fast.ai/index.php/Log_Loss
7. https://towardsdatascience.com/wtf-is-image-classification-8e78a8235acb
8.https://www.researchgate.net/post/What_is_the_pros_and_cons_of_Convolutional_neural_networks
9. https://www.quora.com/What-are-the-differences-among-AlexNet-GoogleNet-and-VGG-in-the-context-of-convolutional-neural-networks