Capstone Project Vedangi Deshpande

Machine Learning Engineer Nanodegree **June 3, 2021**

Project Overview:

In this Dog breed classifier in cubic centimetre the matter statement is to spot the breed of dog, if dog image is given as input, if equipped a picture of an individual's, we've to spot the resembling dog breed. the concept is to make a pipeline of system which will method universe user equipped pictures associate degreed determine an estimate of the canine's breed. this is often a multi-class classification drawback wherever we are able to use supervised machine learning to unravel this drawback.

Problem Statement:

The goal of the project is to create a machine learning model which will be used among internet app to method real-world, user-supplied pictures. The formula has got to perform 2 tasks:

- Dog face detector: Given a picture of a dog, the formula can establish AN estimate of the canine's breed.
- Human face detector: If provided a picture of an individual's, the code can establish the resembling dog breed.

Metrics:

Metrics the information is split into train, check and valid dataset. The model is trained with the train dataset. we have a tendency to use the testing information to predict the performance of the model on unseen information. we are going to use accuracy as a metric to judge our model on check information. Accuracy=Number things|of things} properly classified/ All classified items additionally, throughout model coaching, we have a tendency to compare the check information prediction with validation dataset and calculate Multi category log loss to seek out the most effective playacting model. Log loss takes into the account of uncertainty of prediction supported what proportion it varies from actual label and this may facilitate in evaluating the model.

Data Exploration

For this project, the input format should be of image kind, as a result of we would like to input a picture and establish the breed of the dog. The dataset has photos of dogs and humans. Dog pictures dataset: The dog image dataset has 8351 total pictures that area unit sorted into train (6,680 Images), check (836 Images) and valid (835 Images) directories. every of this directory (train, test, valid) have 133 folders reminiscent of dog breeds. the photographs area unit totally different|of various} sizes and different backgrounds, some pictures don't seem to be full-sized. the information isn't balanced as a result of the amount of pictures provided for every breed varies. Few have four pictures whereas some have eight pictures. Human pictures dataset: The human dataset contains 13233 total human pictures that area unit sorted by names of human (5750 folders). All pictures area unit of size 250x250. pictures have completely different|completely different} background and different angles. the information isn't balanced as a result of we've got one image for a few folks and lots of pictures for a few.

Algorithms and techniques:

For performing arts this multiclass classification, we are able to use Convolutional Neural Network to unravel the matter. A Convolutional Neural Network (CNN) could be a Deep Learning formula which may soak up AN input image, assign importance (learnable weights and biases) to numerous aspects/objects within the image and be able to differentiate one from the opposite. the answer involves 3 steps. First, to observe human pictures, we are able to use existing formula like OpenCV's implementation of Haar feature primarily based cascade classifiers. Second, to observe dog-images we are going to use a pretrained VGG16 model. Finally, when the image is known as dog/human, we are able to pass this image to AN CNN model which is able to method the image and predict the breed that matches the most effective out of 133 breeds.

Benchmark

The CNN model created from scratch should have accuracy of a minimum of 10 percent. This would make sure that the model is functioning as a result of a random guess will give an accurate answer roughly one in 133 times, that corresponds to Associate in accuracy of but 1%.

Data Preprocessing

All the pictures are resized to 224*224, then standardisation is applied to all or any pictures (train, valid and check datasets). For the coaching knowledge, Image augmentation is finished to cut back overfitting. The train knowledge pictures ar indiscriminately turned and random horizontal flip is applied. Finally, all the pictures are reborn into tensor before passing into the model.

Implementation

I have built a CNN model from scratch to solve the problem. The model has 3 convolutional layers. All convolutional layers have kernel size of 3 and stride 1. The first conv layer (conv1) takes the 224*224 input image and the final conv layer (conv3) produces an output size of 128. ReLU activation function is used here. The pooling layer of (2,2) is used which will reduce the input size by 2. We have two fully connected layers that finally produces 133-dimensional output. A dropout of 0.25 is added to avoid over overfitting.

Refinement

The CNN created from scratch have accuracy of thirteen, tho' it meets the benchmarking, the model is considerably improved by exploitation transfer learning. to make CNN with transfer learning, I actually have hand-picked the Resnet101 design that is pre-trained on ImageNet dataset, the design is one hundred and one layers deep. The last convolutional output of Resnet101 is given as input to our model. we tend to solely got to add a totally connected layer to provide 133-dimensional output (one for every dog category). The model performed very well when put next to CNN from scratch. With simply five epochs, the model got eighty one accuracy.

Model Evaluation and Validation

Human Face detector: The human face detector function was created using OpenCV's implementation of Haar feature based cascade classifiers. 98% of human faces were detected in first 100 images of human face dataset and 17% of human faces detected in first 100 images of dog dataset. Dog Face detector: The dog detector function was created using pre-trained VGG16 model. 100% of dog faces were detected in first 100 images of dog dataset and 1% of dog faces detected in first 100 images of human dataset. CNN using transfer learning: The CNN model created using transfer learning with ResNet101 architecture was trained for 5 epochs, and the final model produced an accuracy of 81% on test data. The model correctly predicted breeds for around 680 images out of 836 total images. Accuracy on test data: 81% (680/836).

Justification

I think the model performance is better than expected. The model created using transfer learning have an accuracy of 81% compared to the CNN model created from scratch which had only 13% accuracy.

Improvement

The model can be improved by adding more training and test data, currently the model is created using only 133 breeds of dog. Also, by performing more image augmentation, we can avoid overfitting and improve the accuracy. I have tried only with ResNet 101 architecture for feature extraction, May be the model can be improved using different architecture.

References

1. Original repo for Project - GitHub: https://github.com/udacity/deep-learning-v2- pytorch/blob/master/project-dog-classification/

2. Resnet101: https://pytorch.org/docs/stable/_modules/torchvision/models/resnet.html#resnet101

3. Imagenet training in Pytorch: https://github.com/pytorch/examples/blob/97304e232807082c2e7b54c597615dc0ad8f 6173/imagenet/main.py#L197-L198

4. Pytorch Documentation: https://pytorch.org/docs/master/

5. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neuralnetworks-the-eli5-way-3bd2b1164a53

6. http://wiki.fast.ai/index.php/Log_Los

7. https://medium.com/@samratnilesh098/dog-breed-classifier-using-cnn-63633b956525