Machine Learning Engineer Nanodegree

Capstone Proposal

Abhishek Chaturvedi

May 2020

Dog Breed Classification

Domain Background

CNN have been used to great effect in application such as object classification, scene recognition and other applications. Dog Breed Classification is a very specific and well known application of Convolutional Neural Network. It falls under the category of fine-grained image classification problem, where inter-class variations are small and often one small part of the image considered makes the difference in the classification.

Image classification has uses in most of the industries today. Some of these are the manufacturing industry, IT industry and Medical Science. Dog Breed Classification is especially useful for the animal services which work round the clock to provide shelter, food and families for the animals. Most of the animals which are rescued are dogs, in such a case an app like this which can tell about a dog's breed just by a picture comes in handy. The goal of the project is to identify the breed of a dog whose picture is given as input using Deep Learning techniques.

Problem Statement

The task at hand is to build a CNN model that accepts user-supplied images, and in return performs three tasks:

Dog face detector: The model outputs the breed of the dog, if the image is in fact belongs to a dog.

Human face detector: if the input is a human image, the model will output the breed with which the given human image most resembles with.

Exception case: The model will display a suitable message when the image contains neither a dog, nor a human.

Datasets and Inputs

Both the training and testing datasets comprise of images. The datasets have been provided by Udacity. The input format must be of image type. There are 2 datasets, namely Dog images dataset and Human images dataset.

- **Human Images Dataset :** The dataset contains a total of 13233 human images. These images have different background and angles. All the images are of size 250x250.
- **Dog Images Dataset :** This dataset consists a total of 8351 dog images, which are distributed among 133 different dog breeds. These images are further distributed into train, test and validate set, with 6,680 images in train set, 835 images in valid set and 836 images in test set.

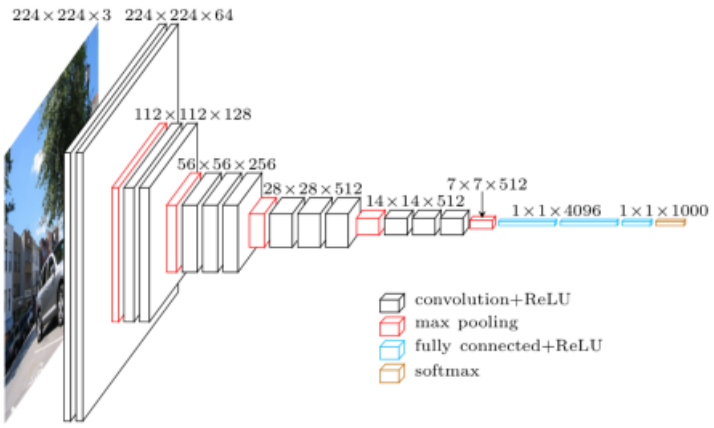
Solution Statement

CNN or Convolutional Neural Networks is a class of Deep neural networks which is most commonly applied to analyze visual imagery. CNNs are regularized versions of multilayered perceptrons. Furthermore, CNNs use relatively little pre-processing compared to other image classification algorithms.

The first step will be to detect human images (using OpenCV's implementation of Haar feature based cascade classifier) and dog images (using pretrained VGG-16 model) after that I will proceed by building a simple CNN with multiple layers each followed by a Max Pooling 2D, from scratch and then I will proceed with fine tuning it with the help of ResNet50 model. I am choosing ResNet50 over VGG16 for transfer learning as Residual networks are very deep compared to VGGs and often result in higher accuracy.

Benchmark Model

For this problem, I will be using VGG-16 as the benchmark model. VGG-16 is a CNN architecture which was used to win ILSVR(ImageNet) competition in 2014.VGG-16 consists of an architecture and weights which have already been trained on ImageNet which is the most popular dataset used for image classification. It makes the improvement over AlexNet by replacing large kernel-sized filters (11 and 15 in the first and second convolutional layer, respectively) with multiple 3x3 kernel - sized filters one after another. VGG16 was trained for weeks and was using NVIDIA Titan Black GPU. Below is the architecture of a standard VGG16 model.



Architecture of VGG16

Evaluation Metrics

I will use Accuracy metric as the evaluation metric. Accuracy is the ratio of number of correct predictions to the total number of input samples. Accuracy works best when the data is balanced. In our case the classes are approximately evenly distributed over training, validation and testing sets and hence we can use accuracy.

Project Design

I have divided the project into 3 parts-

- Pre-processing: In this step I will import all the required datasets and libraries. After that I will pre-process the data.
- Data Splitting: I will split the input data into train, validate and test datasets, furthermore, if required, I will perform image augmentation on training data.
- Model training and evaluation: This step will comprise of the following:
    - Create a dog detector using VGG16 model.
    - Train, test and validate a CNN model built from scratch.
    - Again create a CNN model, but this time using transfer learning with ResNet50, and finally train, test and validate the data on this model.
    - Evaluate the model by display of appropriate messages as output.

References

https://github.com/udacity/machine-learning/blob/master/projects/capstone/capstone_proposal_template.md

https://deeplizard.com/learn/playlist/PLZbbT5o_s2xrfNyHZsM6ufI0iZENK9xgG

https://deeplizard.com/learn/video/4Tcqw5oIfIg

https://pytorch.org/docs/stable/_modules/torchvision/models/resnet.html#resnet50

https://pytorch.org/docs/stable/index.html

https://github.com/pytorch/examples/blob/master/imagenet/main.py