Capstone Project

Kushank Sriraj

Machine Learning Engineer Nanodegree

October 10th, 2020

CNN Project: Dog Breed Classifier

Domain Background

This project is based on Convolutional Neural Network(CNN). The dog breed classification is a very popular problem in deep learning. The project also takes advantage of Transfer Learning and some state-of-the-art CNN models.
These deep learning algorithms are well suited for computer vision or working with images.
My motivation for doing this project is to learn and practically apply the CNN and transfer learning techniques on real-world dataset. After successfully doing this project, I'm planning to make a web app which will take an input image from the user, pass the image to the model and make classifications based on it. I will be using AWS Sagemaker to host my model.

Problem Statement

The problem is to predict the dog's breed given an image of the dog. This is possible by supervised learning. Further, if supplied an image of a human, predict a resembling dog breed. The solution to the problem is quantified by error metrics.

Datasets and Inputs

The datasets : dog dataset[1] and human dataset[2] required for this project is provided by Udacity. The input required is of image type.

More details on datasets :

1. Dog Image dataset : It contains images of various dog breeds. The dataset is not balanced. It contains a total of 8351 images.

2. Human Image dataset : It contains images of humans, sorted by name. This dataset is also not balanced. It contains a total of 13233 images.

Solution Statement

Since this is a problem of image classification, Convolutional Neural Network can be used. CNN is best suited for image classification tasks so it is the perfect choice for this project. Further, for detecting human faces, the Haar Cascade algorithm will be employed.
In order to reach a higher accuracy, a pretrained model(VGG-16) will be used in context with transfer learning.
The solution will be measurable with the help of evaluation metrics stated here.

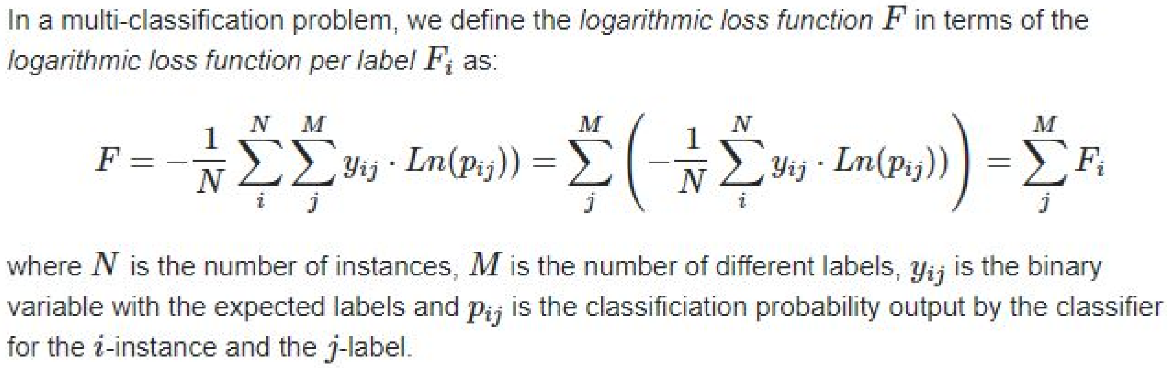
Benchmark Model

On the basis of preliminary research, I found that :

- The CNN model that will be created as a result of transfer learning based on the VGG-16 model should have 60% or greater accuracy.
- The model created without transfer learning should have an acceptable accuracy that is better than a random guess.

Evaluation Metrics

Since this a multi-class classification problem, the Multi-class logarithmic loss function is best suited as an error or evaluation metric.



In a multi-classification problem, we define the *logarithmic loss function* F in terms of the *logarithmic loss function per label* F_i as:

$$F = -\frac{1}{N}\sum_{i}^{N}\sum_{j}^{M} y_{ij} \cdot Ln(p_{ij})) = \sum_{j}^{M}\left(-\frac{1}{N}\sum_{i}^{N} y_{ij} \cdot Ln(p_{ij}))\right) = \sum_{j}^{M} F_i$$

where N is the number of instances, M is the number of different labels, y_{ij} is the binary variable with the expected labels and p_{ij} is the classificiation probability output by the classifier for the i-instance and the j-label.

Images source[3]

Project Design

The workflow of the solution will be as follows:

1. Load the required libraries and the datasets.
2. Do data preprocessing and split the data into train, test and validation sets.
3. Use Haar Cascade algorithm of OpenCV library to detect human face.
4. Load the pretrained VGG-16 model and perform prediction.
5. Create the dog breed classifier CNN model from scratch.
6. Specify loss function and evaluate the CNN model.
7. Create the CNN model using transfer learning, predict and evaluate it.
8. Write an algorithm to predict dog breed from human image.

Reference :

1. https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/dogImages.zip
2. http://vis-www.cs.umass.edu/lfw/lfw.tgz
3. https://stats.stackexchange.com/questions/113301/multi-class-logarithmic-loss-function-per-class
4. https://github.com/udacity/deep-learning-v2-pytorch/blob/master/project-dog-classification/README.md