Machine Learning Engineer Capstone Project

Proposal: Dog Breed Classifier

James Lau

Domain Background

Convolutional neural networks (CNN) have broken the mold and ascended the throne to become the state-of-the-art computer vision technique. Among the different types of neural networks (others include recurrent neural networks (RNN), long short term memory (LSTM), artificial neural networks (ANN), etc.), CNNs are easily the most popular. These convolutional neural network models are ubiquitous in the image data space. They work phenomenally well on computer vision tasks like **image classification**, object detection, image recognition, etc. [1]

A CNN consists of an input layer, output layer, as well as multiple hidden layers. The hidden layers of a CNN typically consist of convolutional layers, pooling layers, fully connected layers, and normalization layers (ReLU). Additional layers can be used for more complex models. This architecture has shown excellent performance in many Computer Vision and Machine Learning problems.

Problem Statement

In this project, we will apply Convolutional neural networks (CNN) to classify the bleed of a dog based on a given image. If the image is neither a human nor a dog, the output will tell us it's not a human or dog. Given an image of a dog, the algorithm will identify an estimate of the canine's bleed. If supplied an image of a human, it will identify the resembling dog breed. We will also use different models for benchmarking to see their strengths and weaknesses.

After completing this project, I can learn how to build a pipeline to process a real-world problem; from data processing, model design, improvements (use transfer learning) to the final model.

Datasets and inputs

I get the input datasets provided by Udacity. One dataset is for the dog images (8351 images), and the other is for human images (13233 images).

Dog images:

https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/dogImages.zip

Human images:
https://s3-us-west-1.amazonaws.com/udacity-aind/dog-project/lfw.zip

The dog images will be divided into 3 groups:

1. Training set
2. Validation set
3. Test set

The dog dataset contains 8351 dog images. it will be split into 6680 images for training, 836 images for testing, and 835 images for validation.

All 3 datasets have 133 dog breeds. The training dataset has an average of 50 images for each breed. For the training and test datasets, it has 8 images for each breed in average. The average number for images for each bleed is quite high and balance.

Solution Statement

Convolutional neural networks (CNN) will be developed to classify dog breeds, and a test accuracy of at least 10% must be attained.

After preprocessing the data, a CNN architecture will be defined, then we will specify the Loss Function and Optimizer to get a better result. After train, validate and test the model, we can see how good(accuracy) the model is. We will then use Transfer Training to attain a higher accuracy of at least 60%. Transfer Learning is a Machine Learning technique whereby a model is trained and developed for one task and is then re-used on a second related task. It refers to the situation whereby what has been learnt in one setting is exploited to improve optimization in another setting . Transfer Learning is usually applied when there is a new dataset smaller than the original dataset used to train the pre-trained model. We will train, validate, and test the model again to get the accuracy after implementing Transfer Learning. [2]

Benchmark Model

The goal of the project is to reach an accuracy of 60% or higher.

Two more models **(ResNet-50** and **Inception-v3)** will be used in this study as a comparison.

The **ResNet-50** (a 50-layer deep-CNN architecture) is the first deep-CNN architecture that utilized residual learning in 2015. ResNet-50 have been successful in increasing accuracy in computer vision benchmarking challenges, winning first prize in the ImageNet Large Scale Visual Recognition Challenge 2015 (ILSVRC, 2015) and Microsoft Common Objects in Context 2015 (MS COCO, 2015), The ResNet-50 model was trained on 1.28 million training images in 1000 classes and reaching an average of 5.25% of top-5 error. Based on the success of ResNet-50 architecture [3]

Inception-v3 is a pre-trained convolutional neural network that is 48 layers deep, which is a version of the network already trained on more than a million images from the ImageNet database. This pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. The model extracts general features from input images in the first part and classifies them based on those features in the second part. Inception v3 is a widely used image recognition model that has been shown to attain greater than 78.1% accuracy on the ImageNet dataset and around 93.9% accuracy in top 5 results. [4]

Evaluation Metrics

Evaluating a model is a core part of building an effective machine learning model. Just building a predictive model is not enough, we need to create and select a model which gives high accuracy; therefore, it is crucial to check the accuracy of the model prior to computing predicted values. In this study, accuracy will be used as an evaluation metric to find the model that's a better pick for a dog bleed classifier.

Project Design

The project design is listed as the followings.

1. Import Datasets
    - The data is downloaded from the site provided by Udacity.
    - For different Pre-defined model, the minimal image size may be different, I will need to resize the image data for different model. The size is scale to 224 initially.
    - I may explore the data augmentation to improve the accuracy.

2. Detect Humans
3. We use OpenCV's implementation of Haar feature-based cascade classifiers to detect faces in images.
4. Detect Dogs

    Use a Pre-trained VGG-16 Model to detect dogs.

5. Create a CNN to Classify Dog Breeds (from Scratch)
    - The architecture I have in mind to implement from scratch (without using transfer learning yet) is VGG16
    - Need to predict breed from images and attain a test accuracy of at least 10%.
    - Train, validate and test the model
    - I will use transfer learning to fine tune the model
6. Create a CNN to Classify Dogs (using Transfer Learning)
    - It must attain at least 60% accuracy on the test set.

- Train, validate and test the model.

References

[1] https://www.analyticsvidhya.com/blog/2020/02/learn-image-classification-cnn-convolutional-neural-networks-3-datasets

[2] https://towardsdatascience.com/build-your-first-computer-vision-project-dog-breed-classification-a622d8fc691e

[3] https://www.researchgate.net/profile/Dmitry-Konovalov-2/publication/328418948_Bird_Call_Recognition_using_Deep_Convolutional_Neural_Network_ResNet-50/links/5bcd62dc458515f7d9d02755/Bird-Call-Recognition-using-Deep-Convolutional-Neural-Network-ResNet-50.pdf

[4] https://www.kaggle.com/imsparsh/large-categories-image-classifier-inception-v3