Machine Learning Engineer Nanodegree

Capstone Proposal

Noel Mathew September 25, 2020

Proposal

Dog Breed Classifier

Domain Background

Computer vision is a field of artificial intelligence that trains computers to interpret and understand the visual world. The idea of building intelligent machines that can understand and interpret visual world existed from 1960s. It all started with Hubel and Wiesel's experiment to understand how a cat's neuron responded to visual stimuli. MIT started a Summer Vision Project in the summer of 1966, with a goal to solve the vision problem in a single summer. Though it was not a success, it led to people getting interested and further pushed the field. Today we have not solved the problem yet, but we have surely come far. Far enough be used with medical imaging, object detection, image captioning and so on. The field was revolutionized with the introduction of Deep Learning and the hardware support provided by Nvidia GPUs and was demonstrated by Alexnet winning the Imagenet Competition in 2012.

My whole journey with Machine Learning started with being amused by what deep learning has made possible in the fields of Computer Vision. I will be using Deep learning to estimate dog breeds which is not a trivial problem to solve due to the interclass variance and the intraclass variance between the dog breeds.

Problem Statement

The goal of the project is to determine which dog breed a given image contains. I will use Convolutional Neural Network (CNN) to classify the images by dog breeds. If the image contains a human then output the resembling dog breed. If the image contains neither of the two then it will report an error. I will be using a custom loss function, a combination of two Binary Cross Entropy loss `dog_loss` & `face_loss` and a Cross Entropy loss for `breed_loss`. The breed_loss will be conditional; i.e., the breed_loss will only be calculated in case the image is actually a dog.

Furthermore, One Cycle Scheduling with cosine annealing for scheduling learning rates and momentums will be used.

Datasets and Inputs

The datasets are provided by Udacity.

Udacity has provided us with two sets:

1. `dog dataset`: The dataset contains 133 folders; each corresponding to a different dog breed. The dataset is split into train, test and valid folders. There 8351 total dog images.
    - **Train dataset**: We have 6680 images in the training dataset. We have an average of 50 images with a minimum of 26 images a breed and a maximum 77 images a breed.

    - **Test dataset**: We have 836 images in the training dataset. We have an average of 6 images with a minimum of 3 images a breed and a maximum 10 images a breed.
    - **Valid dataset**: We have 835 images in the training dataset. We have an average of 6 images with a minimum of 4 images a breed and a maximum 9 images a breed.
    - We have enough images to not have any class imbalance problem with transfer learning.
    - Each image is an RGB image.
2. `human face dataset`: The dataset contains folders corresponding to different people containing photos. There are 13233 total human images. The human images are arrange by names of the people. I won't be using the names of the people respecting their privacy. We will be only using the images for training the face detection classifiers. The images are RGB. I will be splitting the dataset as 70% train set and 15% validation set and 15% test set.

The dog dataset will be used to detect dogs and predict the dog breeds. The human face dataset will be used to detect faces.

Solution Statement

I will be using a Convolutional Neural Network based architecture pre-trained on ImageNet dataset to perform transfer learning. I will train a single model with multiple heads for both tasks (**i.** dog or human face detection **ii.** dog breed classification).

Benchmark Model

- **CNN from scratch** : This benchmark consists of training a CNN from scratch. This is a very basic benchmark and will help to understand if the model is learning. Detecting dog breeds with a random model would have a 1 in 133 chance; i.e. less than 1%. The benchmark is aimed for atleast 10%, making sure that the model is learning.

Evaluation Metrics

- **Detection Accuracy**: This metric gives the accuracy with which we detect dogs and faces.
- **Breed Accuracy**: This metric gives the accuracy for the breeds.

Project Design

I will be using Deep Learning to detect dogs and human faces. I will build a CNN classifier with 135 activations as output; 1 for each dog breed(133) and 1 each for detecting a face or a dog. I will use sigmoid activation to detect the dogs or human faces and use softmax activation on each of the dog breed activations. I will use transfer learning where I will use a model pre-trained on Imagenet Dataset to learn our task.

I will preprocess the dataset with resizing the images to a fixed size of 224x224, use random resized crops and random perspective transformation. This should give me enough variation given the size of the dataset.

I will use Binary Cross Entropy loss `dog_loss` and `face_loss` for the sigmoid activations and a Cross Entropy loss `breed_loss` for the softmax activations This way the network can predict if the image contains a person or a dog or none of those and predicts the dog breed that the network is most confident about. The final loss function would be :

```
final_loss = dog_loss + face_loss + is_dog*breed_loss
```

As described in the problem statement, if the network detects a dog, the dog's breed is returned. If the network detects a human face, the most resembling dog breed according to the network is returned. And if none of both are detected, It could be handled as an Error.

References:

A disciplined approach to neural network hyper-parameters: Part 1 -- learning rate, batch size, momentum, and weight decay : https://arxiv.org/abs/1803.09820

ImageNet Classification with Deep Convolutional Neural Networks : https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf