Machine Learning Engineer Nanodegree

Capstone Proposal

Viswanath Ravindran
April 25th, 2018

Proposal

One of the primary reasons for me to take up the Udacity Advances Machine learning Nanodegree is that it would provide me with a platform to introduce me to Deep Learning. Computer Vision problem being part of the curriculum excited me even further. My knowledge and understanding of Markov Decision Process and Computer Vision has improved several folds after going through the course. Considering the liking that I have acquainted for Computer Vision I would like to do my Capstone project on this specific area. I have chosen the Humpback Whale Identification Challenge @ link from Kaggle

Domain Background

Solving Computer Vision Problem has been explored by several researchers in the recent years. Udacity's introduction to the CIFAR-10 dataset as a problem for the Deep learning project introduced me to attempting to solve Computer Vision problem. There are several research papers published on it till date.

Problem Statement

Problem Statement obtained from Kaggle page @ link

After centuries of intense whaling, recovering whale populations still have a hard time adapting to warming oceans and struggle to compete every day with the industrial fishing industry for food. To aid whale conservation efforts, scientists use photo surveillance systems to monitor ocean activity. They use the shape of whales' tails and unique markings found in footage to identify what species of whale they're analyzing and meticulously log whale pod dynamics and movements. For the past 40 years, most of this work has been done manually by individual scientists, leaving a huge trove of data untapped and underutilized.

In this competition, I am challenged to build an algorithm to identifying whale species in images. I will analyze Happy Whale's database of over 25,000 images, gathered from research institutions and public contributors. By contributing, I will help to open rich fields of understanding for marine mammal population dynamics around the globe.

How this problem is important to me now.

I strongly believe in Data Science for a noble cause and this challenge which will span for the next 3 months will help me work towards the same. It will also further deepen my understanding on the computer vision problem. The intricate complications involved here will also hone my skills in approaching the problem of the same domain.

Datasets and Inputs

I will analyze Happy Whale's database of over 25,000 images, gathered from research institutions and public contributors. By contributing, I would help to open rich fields of understanding for marine mammal population dynamics around the globe.

The datasets details are available @ link

The input train data has several image of the whale's fluke and the corresponding identification numbers for different whale species. The test data has unlabeled picture which is required to be processed and label them as accordingly. This is an extreme Multi Class Classification problem.

Solution Statement

I would be applying Deep Learning Multi class classification to solve the problem at hand. As we generally know that Deep Learning Models are effective in learning features effectively in any given set of images using various distinctions.

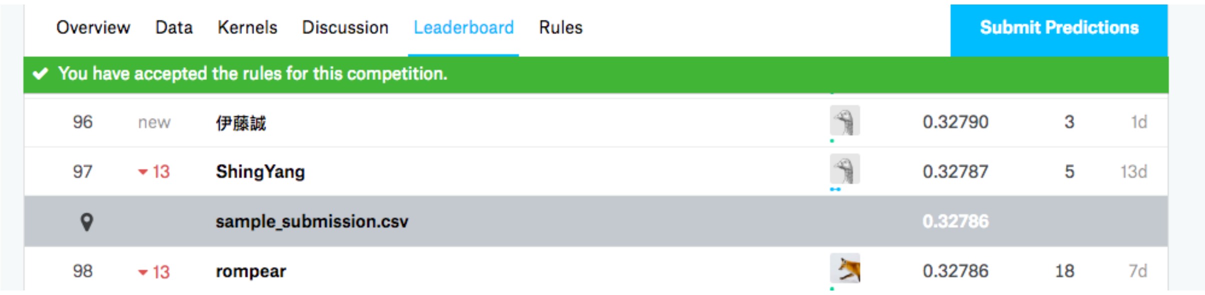
Following initial exploratory Data analysis, I would want to understand the class distribution between the different labels. I would start by applying stacked Convolutional Neural nets (pre trained model) to extract features followed by fully connected layers. I would further improve the model performance applying supporting techniques.

An inspiring paper that I found that talks about the challenges and methods to address them is @ link. This paper is very closely related to the problem that I am trying to address. The various citations that are mentioned in the Bibliography section are quite interesting and should prove to be helpful.

Benchmark Model

The benchmark model that can be used to see how I perform is using the sample submission here. The details of the metric that is used to assess the Model performance is given in detail in the next section. The higher the score indicates a better model. Screenshot attached below shows the Benchmark score submitted using the sample by the host organization of this problem. It is listed in the leaderboard page for the competition @ link

As of the time that I wrote this proposal the base submission is 0.32786. I would focus on working on my model to score better than this.



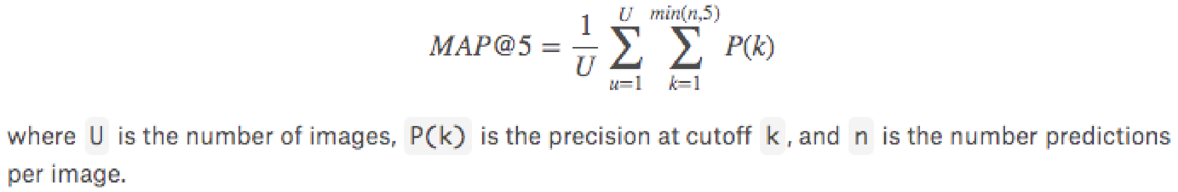
✓ You have accepted the rules for this competition.

96	new	伊藤誠	0.32790	3	1d
97	▾13	ShingYang	0.32787	5	13d
		sample_submission.csv	0.32786		
98	▾13	rompear	0.32786	18	7d

Evaluation Metrics

The evaluation metric for this challenge is called the Mean Average Precision score at K. The k selected here is 5, the output prediction should output the 5 most possible class labels of the several available class in the order of reducing probability.

This measurement metric is popular in Information Retrieval among various Multi class Classification problems setting. The mathematical representation is available as in Kaggle @ link



$$MAP@5 = \frac{1}{U} \sum_{u=1}^{U} \sum_{k=1}^{min(n,5)} P(k)$$

where `U` is the number of images, `P(k)` is the precision at cutoff `k`, and `n` is the number predictions per image.

Project Design

This specific problem can be approach in the following manner:

1. Input the training images and the labels for the training image is available in the necessary format. Images are available in a folder and a single file containing the Labels.
2. After understanding the distribution of the dataset across the various classes that we, I would then have to apply appropriate data split techniques to split between train and validation sets.
3. Identification of potential duplicate images has to be verified since it can skew the class distribution and potentially the model will be unable to learn new features in duplicate images.
4. We can create an initial model as a baseline model using external pre-trained models like ReseNet, VGG16 etc.
5. Using the baseline model and applying various transformation to the image we then can train an inference model to help understand the differences in the predicted vs true label.
6. I would possibly using ImageDataGenerator for real-time data augmentation, layer freezing and model fine-tuning to improve my predictions.
7. We can then fine-tuning the top layers of a pre-trained network. Selecting an appropriate architecture such as the GoogleNet etc is essential for improved accuracy.
8. Choosing an appropriate solver algorithm for better convergence is necessary to be tested.

The final submission file should predict the top 5 predicted class against each input image from the test set. The various submissions will be listed in the Kaggle page which will included in the final project report.