Machine Learning Engineer Nanodegree
Capstone Proposal

Sarthak Sahu
26 June 2018

Invasive Species Monitoring: Identify images of invasive hydrangea (Kaggle Competition)

Domain Background

Tangles of kudzu overwhelm trees in Georgia while cane toads threaten habitats in over a dozen countries worldwide. These are just two invasive species of many which can have damaging effects on the environment, the economy, and even human health. Despite widespread impact, efforts to track the location and spread of invasive species are so costly that they're difficult to undertake at scale.

Currently, ecosystem and plant distribution monitoring depends on expert knowledge. Trained scientists visit designated areas and take note of the species inhabiting them. Using such a highly qualified workforce is expensive, time inefficient, and insufficient since humans cannot cover large areas when sampling.

Because scientists cannot sample a large quantity of areas, some machine learning algorithms are used in order to predict the presence or absence of invasive species in areas that have not been sampled. The accuracy of this approach is far from optimal, but still contributes to approaches to solving ecological problems.

Techniques from computer vision alongside other current technologies like aerial imaging can make invasive species monitoring cheaper, faster, and more reliable.

Problem Statement

We are required to develop algorithms to more accurately identify whether images of forests and foliage contain invasive hydrangea or not. The problem requires training a model to do the classification. The problem is a standard image classification problem wherein we will be given labelled images to train our model.

Datasets and Inputs

The data set contains pictures taken in a Brazilian national forest. In some of the pictures there is *Hydrangea*, a beautiful invasive species original of Asia. Training pictures have labels which should be used to train model and testing set contains pictures on which prediction is to be made.

File descriptions
- train.7z - the training set (contains 2295 images).
- train_labels.csv - the correct labels for the training set.
- test.7z - the testing set (contains 1531 images), ready to be labeled by your algorithm.
- sample_submission.csv - a sample submission file in the correct format.

Data fields
- name - name of the sample picture file (numbers)
- invasive - probability of the picture containing an invasive species. A probability of 1 means the species is present.

Link to dataset :- https://www.kaggle.com/c/invasive-species-monitoring/data

Solution Statement

A solution to the problem is training a deep neural network model with the training images. The deep neural network can be Artificial Neural Networks or Convolutional Neural Networks. CNNs are better at classifying images so, it is likely that the solution proposed will have a CNN to perform classification. Perhaps, transfer learning can be used later on to improve the performance of the model. In transfer learning we use pre-trained model weights on similar kind of problem to solve our problem. One example is inception V3 model which can be used for transfer learning.

Benchmark Model

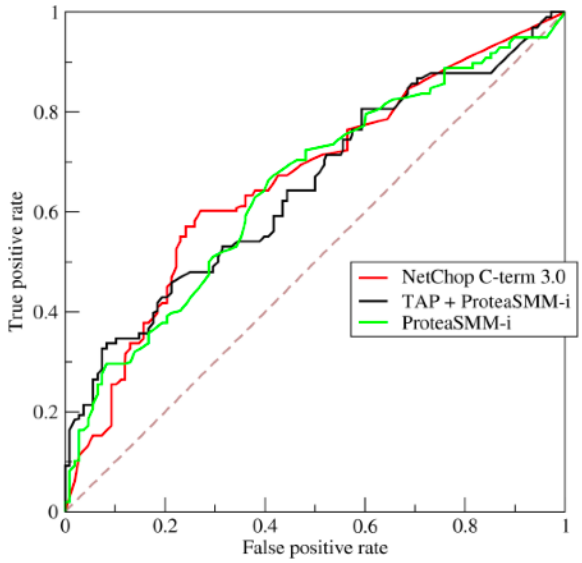
A benchmark model is a simple 2 convolutional layer CNN with max pooling and fully connected layer. Such a model is a good starting point for image classification tasks. I will make a submission using the model on Kaggle and take the score and rank received as benchmark. I will then try to improve by applying successive better approaches.

Evaluation Metrics

Submissions are evaluated on Area under ROC curve between the predicted probability and the observed target. The ROC curve is a graphical plot that illustrates the performance of any binary classifier system as its discrimination threshold is varied. When we have a binary classifier we get output either 0 or 1. We give a threshold around which we give classification into one of the classes. The performance of the classifier varies as we change the threshold.

So, to plot the ROC curve we plot true positive rate (y axis) vs false positive rate (x axis) at various threshold settings. As we vary the thresholds, we get a curve. The area under the curve gives an idea of how good the classifier is.

True positive rate = Correctly Classified Positives/(Correctly Classified as Positives+ Falsely Classified as Negatives)
False positive rate = Incorrectly Classified as Positives/(Incorrectly Classified as Positives+ Correctly classified as Negatives)



In our case the ROC curve will be drawn between predicted probability of invasive species vs actual observation.

Project Design

1. Importing the datasets for the invasive species monitoring from Kaggle.
2. Doing some dataset exploration like the image quality, labels, dataset balance, visualisation etc.
3. Preprocessing stage: separating test, training and validation datasets, doing normalisation, etc.
4. Creating a simple model for benchmark using 2 convolution layers, 2 max pool layers, dropouts and fully connected layers.
5. Train the network and test its performance in test sets and submit for getting a score in Kaggle leaderboard.
6. Tune the architecture and parameters for getting optimal performance with the simple model.
7. Use a pre-trained neural network model and do parameter tuning on it to further improve the performance. I will try more than one model and get the best out of them. This step will continue till I get some visible performance boost compared to the benchmark model.
8. Make submission on Kaggle to check if it performed better than the benchmark model based on evaluation metric.

References

1. https://www.kaggle.com/c/invasive-species-monitoring
2. https://www.quora.com/Whats-ROC-curve
3. Transfer learning - Andrew Ng lecture - https://www.youtube.com/watch?v=yofjFQddwHE