Machine Learning Engineer Nanodegree

Capstone Proposal

Poisonous Plant Classifier

Divya Amin

April 17th, 2019

Domain Background

Poisonous plants are those plants that produce toxins that deter herbivores before eating them. Plants cannot move so they have to protect themselves from other animals in other ways. The most common form of protection for plants is chemical. The poison produced by plants is usually meant to repel insects, but humans and other animals occasionally ingest or come into contact with the poisonous parts of the plants. As a result they suffer from a variety of symptoms such as mild discomfort, nausea, diarrhea etc. But poisonous plants can cause fatal effects such as send a person into a coma, irregular heartbeats, convulsions etc. These plants can affect the lungs, digestive system, nervous system, heart as well as other parts in small amounts. In the cases of plants such as Daphne and Castor beans, a single seed(in the case of the Castor beans) or a few berries of Daphne can kill a person.

These poisonous plants can be mistaken for normal plants and so humans, especially children mistake them for a benign plant and may consume or touch the plants. Many humans except those who have jobs or hobbies that have them come across these plants on a regular basis such as forest rangers, hikers, people who work in nurseries or in maintenance of botanical parks, do not have knowledge about what these plants look like, the minor differences between the poisonous plants and their benign lookalikes. According to the National Capital Poison Center in the US, there were 660 poison exposures reported per 100,000 population. This machine learning algorithm can be used to identify a poisonous plant so that people who do not have prior knowledge of how these plants look can avoid or prevent other from eating or touching those plants.

Problem Statement

1. **Description:** Building a poisonous plant classifier to identify poisonous plants in a picture.
2. **Challenge:** Many children and adults have little to no knowledge of poisonous plants and so can be harmed if they ingest or come into contact with the plants. Poisonous plants can look like normal benign plants and have a higher chance of being misidentified.

3. **Solution:** A Poisonous Plant Classifier that uses a Deep Convolutional Neural Network to identify the poisonous plant in the image uploaded. This problem is measurable in terms of accuracy and loss.

Datasets and Inputs

The dataset to be used in the project is taken from Kaggle(link no. 3 in references). This dataset contained images of poisonous plants such as wisteria, oleander, lily of the valley etc. These images were obtained from Here the images were segregated into a training set, a validation set and a testing set. This helps in training the model. The input will be a image of a poisonous plant. The model will identify the plant in the picture and will give the name of the poisonous plant as an output.

Solution Statement

The solution that will be considered will be to use the Resnet50 Deep Convolutional Neural Network architecture. This is because Resnet50 uses residual learning, where we attempt to learn the subtraction of feature from the input of a Neural Network layer instead of trying to learn some features at the end of a layer. Resnet50 does this by using shortcut connections i.e directly connecting the output of nth layer to the (n+x)th layer. In the case of most neural networks as we go deeper(add more layers) the training of the network takes longer and the accuracy starts degrading as well. Resnet50 is much easier to train and solves the problem of degrading accuracy as well.

Benchmark model

The benchmark model(given in link 4 of References) uses a Resnet18 model. As the model is a classification model, the metric used in the model is accuracy. This model, like the one proposed here is also used to identify which poisonous plants are recognized in the images.

The model is measured using the metrics accuracy and loss. This Resnet18 model has an accuracy of 93.39622497558594% and a loss of 0.24084387719631195. In one picture the model predicted that the picture was of an oleander when it was actually a foxglove. So this solution will be compared with the proposed solution on the basis of loss and accuracy.

Evaluation Metrics

The evaluation metrics that will be considered for quantifying the model are accuracy and loss. Loss is the summation of errors made for each example for training or validation sets and is usually measured as negative log-likelihood for classification models like the one proposed here, which is given by:

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}\log(\hat{y}^{(i)})$$

Accuracy is ratio of the number of test samples predicted correctly to the total number of test samples. These metrics tell us whether the model is classifying plants correctly or not.

Project Design

The workflow is as follows:

1. Load the training and validation images. The images in the dataset given do not require preprocessing.
2. Explore the data by checking the classes and using show_batch() to display some of the poisonous plants.
3. Training the Resnet50 model on the training data and logging the training/validation loss and accuracy and plotting the confusion matrix for result interpretation.
4. Fine tune the network and train again if accuracy not high enough.
5. Make the network predict on the test data and logging the accuracy and loss on that data.
6. Save and freeze the neural network.

   I will mainly be considering the Resnet50 model to solve the problem because as mentioned earlier it is easy to train and solves the accuracy degradation problem as well.

References

1. https://www.poison.org/poison-statistics-national-data-from-2016
2. https://aggie-horticulture.tamu.edu/earthkind/landscape/poisonous-plants-resources/common-poisonous-plants-and-plant-parts/
3. https://www.kaggle.com/nitron/poisonous-plants-images
4. https://www.kaggle.com/nitron/poisonous-plant-classifier-renset18
5. https://stackoverflow.com/questions/34518656/how-to-interpret-loss-and-accuracy-for-a-machine-learning-model