Plant Seedling Classification

Capstone: Machine Learning Engineer Nanodegree

Manvindra Singh
January 31st, 2018

Proposal

Domain Background

Can you differentiate a weed from a crop seedling? Many weed seedlings can be mistaken for crop seedling as they look alike, it becomes hard to differentiate weed from crop. They can be distinguished when they become mature plant but till then soil quality and crop yield will be impacted. For example, Carrot shows relatively slow development in early growth stages. Hence, competing weeds may overtake the crop plant and limit its access to resources such as sunlight, moisture, and nutrient. Thus, weeds may cause major yield losses, if uncontrolled. Automatic plant species identification process will help in right amount or application of pesticides, fertilization and harvesting of different species on-time to improve the production processes of food industries.

Even many seedlings looks alike each other. Example


Seedling Wild Turnip


Seedling Oil Seed Rape


Seedling Charlock

Wild Turnip can be mistaken for Charlock, which has more grass-green true leaves with many hairs. With Oil-seed Rape, which has almost smooth true leaves. Also, with Wild

Radish, which has more grass-green true leaves with many hairs. In all crops in which this plant is found it is an injurious weed, which causes losses.

All the above weed looks alike to Shepherd's-purse a member of the mustard family. Shepherd's purse has been used as a medicinal herb often recommended as a treatment for both internal and external bleeding.

Recently planted Shepherd's purse Plant

Mentioned weed and Shepherd's purse belong to same Mustard Family, so It becomes difficult to distinguish. The ability to distinguish effectively can mean better crop yields and better stewardship of the environment. The goal of this project is to classify seedling images into species.

Historically, similar project has been carried out.

1. Plant classification using convolutional neural networks (http://ieeexplore.ieee.org/document/7577698/?part=1): They used CNN and compare the result with SVM classifiers to predict Plant Type accurately from the leaves of the plants.

2. Deep-Plant: Plant Identification with Convolutional Neural Networks (https://arxiv.org/pdf/1506.08425.pdf) : Classify images of plants into 44 different species. They used Multi layer perceptron to achieve accuracy of 99.6%.

3. Hybrid Generic-Organ Convolutional Neural Network for Multi-Organ Plant Classification (http://cs-chan.com/doc/ICIP_CR.pdf): Unlike above project, which

only uses leaves, this one used plant organs like flowers, stem, leaves and fruits to classify plants.

2. Problem Statement

This problem is based on Kaggle Competition called “Plant Seedlings Classification”. (https://www.kaggle.com/c/plant-seedlings-classification).

As said earlier, many seedlings of crops look alike another crops. Its difficult to distinguish them with naked eye even for experienced farmers. Each seedling can have different growth rate, some may become larger than other seedling in short time. Seedling from a family may resemble other plant seedling in different growth stage. Maybe 4 months old seedling of a plant resemble 2-month-old seedling of another plant. Attempt here would be to classify seedlings images into its plant species.

Below are few images from seedling for each species in dataset

Maize

Common wheat

Sugar beet

Scentless Mayweed

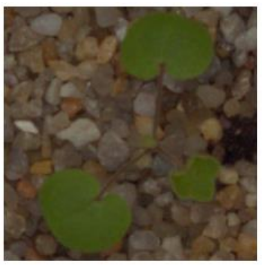
Chickweed

Shepherd's Purse

Cleavers

Charlock

Fat Hen



Cranesbill

Black-grass

Loose Silky-bent

Some of the plant seedlings are similar in appearance. Shepherd's purse is used as herbal medicine whereas it looks alike charlock is a common weed of cornfields.

3. Datasets and Inputs

The Aarhus University Signal Processing group, in collaboration with University of Southern Denmark, has released this dataset containing plants belonging to 12 species at several growth stages. I obtained the dataset through Kaggle which contains 4750 images in training set and 794 in test set. Number of images varies across species and a the size of images varies from 49 X 49 to 3457 X 3991 but most images have aspect ratio of 1.0

List of species along with number of images given in training set is in the following table

Species	Number of Images
Sugar beet	385
Loose Silky-bent	654
Scentless Mayweed	516
Maize	221
Shepherds Purse	231
Cleavers	287
Charlock	390
Small-flowered Cranesbill	496
Fat Hen	475
Common Chickweed	611
Black-grass	263
Common wheat	221
Total	**4750**

4. Solution Statement

Deep learning techniques have been very popular and has outperformed traditional approaches in model performance, feature extractions. They have been used to win one biggest image classification competition -ImageNet. Convolutional neural networks(CNN) unlike other deep learning architect are designed to handle spatial information in images very well. I will use CNN to categories seedling into species. Will also perform data augmentation and apply transfer learning from model trained on ImageNet dataset.

5. Benchmark Model

No benchmark is mentioned in Kaggle competition page. But we can still consider better than random chance of 1/12=0.08 as benchmark. But that is too less for any practical use. So, I have developed a simple CNN model with Three convolutional layers with ReLU activation each following with max pooling layers. I achieved mean F score of 0.657 and accuracy of 0.66. This accuracy is better than random chance of 1/12. I will use this basic CNN model with Mean F score of 0.66 as benchmark model.

6. Evaluation Metrics

1) Micro-Averaged F1-score

F-score is weighted average of precision and recall given by following equations

$$Precision_{micro} = \frac{\sum_{k \in C} TP_k}{\sum_{k \in C} TP_k + FP_k}$$

$$Recall_{micro} = \frac{\sum_{k \in C} TP_k}{\sum_{k \in C} TP_k + FN_k}$$

Here
k is class
TP: True Positive, case where correct species predicted
FP: False Positive, where seeding incorrectly classified to a species
FN: False Negative, cases where seedling is not classified to its species

Once we have these, Mean F score calculated as

$$MeanFScore = F1_{micro} = \frac{2Precision_{micro}Recall_{micro}}{Precision_{micro} + Recall_{micro}}$$

2) Accuracy:
Number of correctly predicted seedling images/ total number of all images in dataset. For this project aim would be to achieve accuracy better than 50%.

7. Project Design

- Programming language: Python 3.6+

- Libraries: Keras, Tensorflow, Scikit-learn, Opencv

Workflow:
4,750 training images are provided to us. Before any preprocessing, these will be divided in train set(80%), validation set(10%) and test set(10%). Test set provided by Kaggle will be used to another hold-out test and will only be used to check if model performed better than benchmark defined.

After the splitting of dataset, preprocessing on images will be done. Images have different size with 3 channels(RGB) so I will resize the image so that they have same width and height. Will try 256px vs 256 px if running smoothly on laptop, If not will decrease the pixels size. Reshape image will have shape (Number of images, 3, 256,256).

Then I will train basic CNN. Starting with 16 locally connected convolutional layers(Filter) followed by Max Pooling layer to down sample parameters. Then added second set of 32 Filters with size 2 followed by Max Pooling Layer. Then 64 Filters again followed by Max Pooling layers. Dropout Probability of 20% to be used on last Conv Layer.
Then Flatten the features set.After this is final layers with 12 nodes to categorize seedling to species and uses Softmax activation.

```
model = Sequential()
model.add(Conv2D(filters=16, kernel_size=2, padding='same', activation='relu', input_shape=(256, 256, 3)))
model.add(MaxPooling2D(pool_size=2))
model.add(Conv2D(filters=32, kernel_size=2, padding='same', activation='relu'))
model.add(MaxPooling2D(pool_size=2))
model.add(Conv2D(filters=64, kernel_size=2, padding='same', activation='relu'))
model.add(MaxPooling2D(pool_size=2))
model.add(Dropout(0.3))
model.add(Flatten())
model.add(Dense(12, activation='softmax'))
model.summary()
```

I will improve the model by

1. Data Augmentation
2. Making CNN denser: add more layers
3. Making CNN wider: add more filters
4. Using different optimization function

Then I will also use transfer learning by using pretrained model like ResNet50, VGG16, InceptionV3 and Xception. Finally, I will use the best model to predict images in test set and post the results on Kaggle to check if I achieved better score than benchmark