Predicting Plant Traits Using Convolutional Neural Networks and Ancillary Data

Aditya Batra
School of Computer Science
University of Waterloo
Waterloo, ON, N2L 3G1
a8batra@uwaterloo.ca

Abstract

This project focuses on predicting specific plant traits by leveraging Convolutional Neural Networks (CNNs) and ancillary environmental data derived from citizen science plant photographs. The model aims to predict six key plant traits. The dataset utilized includes images from the iNaturalist database paired with curated plant trait data from the TRY database, along with ancillary information such as local climate conditions, soil properties, and satellite data. The CNN model employed is a fine-tuned VGG16, pre-trained on the "imagenet" dataset. The ancillary data undergoes minmax normalization, and the target traits are log10 transformed before further normalization. The model's performance, assessed via the R^2 score, and aims to discover the potential of combining image data with environmental information for complex ecological predictions.
Refer to github/adityabatra/PredictingPlantTraits for the code.

1 Introduction

This project builds on the work of Schiller et al. (2021), who demonstrated that Convolutional Neural Networks could predict certain plant traits from photographs. While their study was limited to a smaller dataset and a few traits, this project seeks to extend these capabilities. This project uses a dataset created by combining plant trait data from the TRY database with plant photographs from the iNaturalist database. Each plant photo is linked to its traits based on species names and supplemented with environmental data derived from various global sources. These environmental factors include temperature and precipitation data from WORLDCLIM, soil properties from the SOIL dataset, satellite measurements from MODIS, and radar data from VOD, which helps in understanding plant water content and biomass. By combining this diverse information, the project aims to improve the accuracy of plant trait predictions.

2 Related Works

Relevant literature involves the work done by Schiller et al. (2021), that used Convolutional Neural Networks (CNN) to predict certain plant traits. They combined the output of the CNN with some ancillary data of 6-8 dimensions to predict specific plant traits. Another study by Sharma et al. (2021), demonstrated the power of using VGG16 and Resnet34 CNN to predict plant diseases with an accuracy of 97.58% and 97.77%. According to Carlo et al. (2022), VGG had an accuracy of almost 90.58% after freezing the first 7 layers for plant classification. These related works make it easier for us to choose an appropriate Convolutional Neural Network (CNN) for transfer learning.

Submitted to CS480/680, 2024 Spring. Do not distribute.

3 Methodology

3.1 Problem Formulation

The primary objective of this project is to predict six specific plant traits using a combination of Convolutional Neural Networks (CNNs) and ancillary environmental data. The traits targeted in this study are:

- **X4:** Stem specific density (SSD) or wood density (stem dry mass per stem fresh volume)
- **X11:** Leaf area per leaf dry mass (specific leaf area, SLA or 1/LMA)
- **X18:** Plant height
- **X26:** Seed dry mass
- **X50:** Leaf nitrogen (N) content per leaf area
- **X3112:** Leaf area (in case of compound leaves: leaf, undefined if petiole in- or excluded)

The model was trained on a dataset comprising over 43,000 images and their associated ancillary data, with a validation split of 20% to monitor performance

3.2 Data preprocessing

To ensure that the input data was on a comparable scale, minmax normalization was applied to the ancillary data. This normalization process scales each feature to a [0, 1] range, reducing the potential impact of features with larger ranges. Additionally, the target trait values were log10 transformed before applying minmax normalization.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

where x is the original feature value, $\min(x)$ and $\max(x)$ are the minimum and maximum values of the feature, and x' is the normalized value.

The images were passed through VGG16's preprocessing function which includes normalizing all the pixels by a factor of 1/255.

3.3 Optimizer

Adam optimizer was used with a learning rate of 1e-4. The loss function used was Mean Squared Error.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

where y_i represents the true value, $\hat{y}_i$ represents the predicted value, and n is the number of samples.

Algorithm 1: Adam optimizer

Input: Learning Rate ϵ, Decay rates ρ_1, ρ_2, θ, δ
Output: Optimized parameters θ
Initialize: $s = 0$, $r = 0$, $t = 0$
while *stopping criteria not met* **do**
  Sample example $(x^{(i)}, y^{(i)})$ from training set
  Compute gradient estimate: $\hat{g} \leftarrow \nabla_\theta L(f(x^{(i)}; \theta), y^{(i)})$
  $t \leftarrow t + 1$
  Update: $s \leftarrow \rho_1 s + (1 - \rho_1)\hat{g}$
  Update: $r \leftarrow \rho_2 r + (1 - \rho_2)\hat{g} \odot \hat{g}$
  Correct Biases: $\hat{s} \leftarrow \frac{s}{1-\rho_1^t}$, $\hat{r} \leftarrow \frac{r}{1-\rho_2^t}$
  Compute Update: $\Delta\theta \leftarrow -\epsilon\frac{\hat{s}}{\sqrt{\hat{r}}+\delta}$
  Apply Update: $\theta \leftarrow \theta + \Delta\theta$

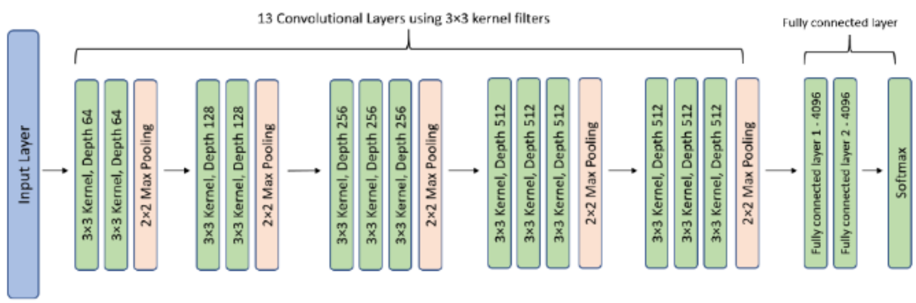
3.4 Model

The model architecture is composed of three main components: the VGG16 CNN for processing image data, a series of dense layers for processing ancillary data, and a combined layer that integrates the outputs of both streams to produce the final predictions.

3.4.1 VGG16 Convolutional Neural Network

The image data is processed using a VGG16 model pre-trained on the "imagenet" dataset. The first 7 layers of the VGG16 model are frozen to retain the learned features while the remaining layers are fine-tuned to adapt to the specific task of plant trait prediction. The VGG16 architecture is composed of the following layers:

- **Input Layer:** Accepts images of size 128x128x3.
- **VGG16 with first 7 layers frozen**



- **Global Average Pooling:** A global average pooling layer is applied to reduce the dimensionality of the feature maps before combining them with the ancillary data.

3.4.2 Ancillary Data Processing

The ancillary data, which includes climate, soil, and satellite-derived metrics, is processed using a series of fully connected layers:

- **Input Layer:** Accepts 163-dimensional input corresponding to the ancillary data features.
- **Dense Layer 1:** A fully connected layer with 128 units, ReLU activation, and L2 regularization.
- **Dropout Layer:** A dropout layer with a 20% dropout rate.
- **Dense Layer 2:** Another fully connected layer with 64 units, ReLU activation, and L2 regularization.
- **Dropout Layer:** A second dropout layer with a 20% dropout rate.
- **Dense Layer 3:** A final fully connected layer with 32 units, ReLU activation, and L2 regularization.

3.4.3 Combined Layers

The outputs from the VGG16 model and the ancillary data processing layers are concatenated to form a combined feature representation, which is then processed through additional fully connected layers to produce the final trait predictions:

- **Concatenation Layer:** Combines the outputs of the VGG16 model and the ancillary data processing layers.

- **Dense Layer 1:** A fully connected layer with 128 units, ReLU activation, and L2 regularization.
- **Dropout Layer:** A dropout layer with a 20% dropout rate is applied.
- **Dense Layer 2:** A fully connected layer with 64 units, ReLU activation, and L2 regularization.
- **Dropout Layer:** A second dropout layer with a 20% dropout rate.
- **Dense Layer 3:** A final fully connected layer with 16 units,ReLU activation, and L2 regularization.
- **Output Layer:** A fully connected output layer with 6 units, corresponding to the six plant traits being predicted. The output activation function is linear.

4 Results and Conclusion

The results are evaluated using the R^2 score, also known as the coefficient of determination. The R^2 score is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

We use a learning rate of 1e-4 with a batch size of 32 and running it for 20 epochs with early stopping. In this project, the R^2 scores for different CNN models are reported, both for the training and validation datasets. The table below summarizes these results:

Table 1: R^2 Scores Model

Metric	Training	Validation
R^2	0.3472	0.21
MSE	0.01538	0.02024

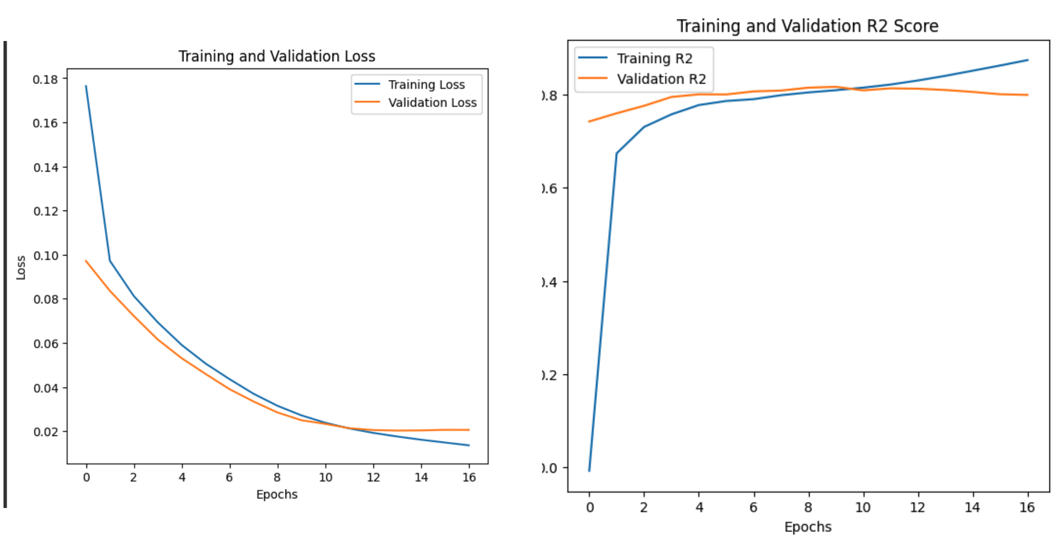


Figure 1: Loss and R^2 during training

Conclusion: The results indicate that the integration of Convolutional Neural Networks with ancillary environmental data can effectively predict multiple plant traits with reasonable accuracy and R^2 value. Further refinement is needed to capture the variance in the data. However, this study shows how Convolutional Neural Networks (CNNs) can be combined with ancillary data and accurately predicy ecological traits using transfer learning.

Acknowledgement

Thanks to the University of Waterloo, Cheriton School of Computer Science, CS480 staff, and professor Yaoliang Lu for their help and guidance throughout the term. This project is inspired by Schiller et al. (2021), and the choice of CNN is influenced by Sharma et al. (2021) and Carlo et al. (2022). The VGG16 model architecture image is extracted from Tammina et al. (2019).

References

Greca, M. L. (2021). “Classification Of Plants Using CNN”. Accessed: 2024-08-12.

S, H., N. Sharma, S. Yerramaneni, C. Santoshini, R. Durga, and V. Akhila (Dec. 2021). “Plant disease prediction using convolutional neural network”. *EMITTER International Journal of Engineering Technology*, vol. 9, pp. 283–293.

Schiller, C., S. Schmidtlein, C. Boonman, A. Moreno, and T. Kattenborn (Aug. 2021). “Deep learning and citizen science enable automated plant trait predictions from photographs”. *Scientific Reports*, vol. 11, p. 16395.

Tammina, S. (Oct. 2019). “Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images”. In: vol. 9, p9420.