DS6050 Project Literature Review II
Toxic Plant Classification

GitHub: Toxic Plant Classification

Katherine Kelleher
School of Data Science
University of Virginia
Charlottesville, USA
kbk8vh@virginia.edu

Margaret Lindsay
School of Data Science
University of Virginia
Charlottesville, USA
jsk2zs@virginia.edu

Brian Nolton
School of Data Science
University of Virginia
Charlottesville, USA
frv3fp@virginia.edu

***Abstract*—Poisonous plants present health hazards to individuals who spend time outdoors across diverse geographic regions. While experts can often identify poisonous plants, many individuals remain vulnerable to the misidentification of toxic vegetation. Drawing on a wide array of literature related to current machine-learning methods and toxic plant classification, this paper covers preliminary experiments that leverage publicly available data to classify images of plants by toxicity status. Both random forest and deep-learning models were trained to identify vegetation that pose health risks and assessed to determine whether they could be used to prevent negative encounters with toxic vegetation.**

I. Introduction

Poisonous plants growing around the world pose not only a significant health risk but also an economic threat to communities. While the old saying "leaves of three, let them be" may be useful for children in the continental United States to avoid an unfortunate encounter with poison ivy in the backyard, more sophisticated means of identifying plants from a wide range of regions, particularly whether a plant is poisonous, can protect individuals from physical harm and economic loss.

Botany experts have advanced methodologies for identifying a plant and whether it is poisonous, non-expert outdoors people and farmers would benefit significantly from having an accessible, non-destructive method of identifying plants they encounter as poisonous or not poisonous.

The motivation for this project is to apply modern machine-learning models to the identification of common plants that are poisonous to humans. This project aims to enable individuals with the capability to identify whether a plant captured within a photo is poisonous or not by utilizing advanced deep learning architectures and techniques.

II. Literature Review

A. Summary of Existing Literature Covered in Literature Review I

Given the potentially grave consequences of misidentifying a poisonous plant, numerous studies exist that seek to improve on traditional plant identification methods by using machine learning. Traditional methods of identifying plants can be time-consuming, requiring years of experience to develop expertise [1]. Machine learning computer vision models offer a faster alternative that would allow non-experts to safely and accurately classify plants as poisonous or non-poisonous. Several studies provide inspiration for the project in this paper and are discussed in Literature Review I (and cited in the references below).

B. Expanded Literature Review

As Convolutional Neural Network (CNN) models can be computationally expensive and time consuming to train, this project includes a random forest baseline model. In their paper, "Image-based yield prediction for tall fescue using random forests and convolutional neural networks," Ghysels, De Baets, et al. provide inspiration for predicting plant matter by evaluating random forest and CNN architectures. [9] Their findings included that the overall performance metrics of the CNN and RF models only differed slightly – no difference in the F1-measure. [9] The CNN method showed superior predictive power for Classes 1 and 2 but performed worse than the RF model for Class 3. [9] These findings indicate that there is value in assessing whether a random forest model is sufficient for the classification of vegetation.

C. Gaps in Existing Literature

The studies considered in Literature Review I and II all resulted in accuracy scores of 92% and above, which suggests neural networks, specifically CNNs, can be used to develop highly successful models for classifying plants as poisonous or non-poisonous.

While all studies reviewed for this project use neural networks to develop computer vision models that classify a plant as poisonous or not, each study uses a different architecture to accomplish the task. There appears to be a need to attempt to reproduce the results of these studies and evaluate model performance on a range of plant species from different regions.

Many of the papers considered did not discuss in detail the process for how the suggested model was developed in terms of tuning hyperparameters and inclusion of components. A

more detailed investigation into hyperparameter optimization and an ablation study could be useful to determine if any further improvements could be made to the accuracy and efficiency of the suggested models.

III. Methodology

A. Research Question

How does the choice of machine-learning architecture and hyperparameter tuning affect the accuracy of poisonous versus non-poisonous plant classification across diverse species and backgrounds?

B. Dataset

This project utilized the Toxic Plant Classification dataset found on Kaggle [8]. This dataset contains nearly 10,000 images of 10 different classes of plants: five which are toxic and five which are not toxic but are often mistaken for a toxic variety. Each class of plant has 1000 images (a couple of them a few less). The dataset is balanced, with roughly equal numbers of toxic and non-toxic plants. The toxic plant classes are western poison oak, eastern poison oak, western poison ivy, eastern poison ivy, and poison sumac. The nontoxic plant classes are Virginia Creeper, Boxelder, Jack-in-the-pulpit, Bear Oak, and Fragrant Sumac. The images were scraped from the iNaturalist site. The dataset includes the slang name (e.g., poison sumac), the scientific name (e.g., Toxicodendron vernix) and its Herbarium 2022 Category ID for each image.

The dataset was split into training, validation, and test samples. In this case 6,966 (69%) of images were kept for model training, 1,493 (15%) for validation, and 1,493 (15%) for were withheld for testing. The dataset split was stratified by species and toxicity, so the training, validation, and test datasets contain an even distribution of toxic and non-toxic plants and an even distribution of species.

Toxicity	Plant	Training	Validation	Testing
0	Bear Oak	700	150	150
0	Boxelder	700	150	150
0	Jack-in-pulpit	700	150	150
0	VA creeper	700	150	150
0	Fragrant Sumac	699	150	150
1	E. Poison Ivy	700	150	150
1	W. Poison Ivy	700	150	150
1	W. Poison Oak	700	150	150
1	Poison Sumac	699	150	150
1	E. Poison Oak	668	143	143

TABLE I: Training Dataset Distribution

C. Methodology

In this project, two main models were investigated for toxic plant classification. The first was a Random Forest model baseline, a traditional machine-learning approach trained to assess whether a deep-learning approach was justified. This model required extracting informative features, including color, texture, and shape. The Random Forest model and was evaluated for accuracy, precision, recall, and F-1 score to determine if it was sufficiently able to classify toxic and non-toxic plant images with a reasonable accuracy. The second was a CNN model, a deep-learning method for classifying images that computes spatial relations between pixel values and their context to extract meaningful features. The base architecture for the model was EfficientNetB0, which was tested with multiple variants and was evaluated for accuracy. Confusion matricies were also analyzed to understand rates of misclassifcation. The following model variants were tested:

- Random forest
- Tuned random forest
- EfficientNetB0, random weights
- EfficientNetB0, CBAM, random weights
- EfficientNetB0, pretrained weights
- EfficientNetB0, CBAM, pretrained weights

IV. Preliminary Experiments

A. Random Forest Classification

In order to test whether a CNN model was necessary to the project's goal of classifying toxic and non-toxic plants, a random forest baseline model was used to assess if a deep learning model should be tested. Features were extracted from the RGB values of the image for the algorithm to use to divide observations based on their features for classification. These included Grey-Level Co-Occurrence Matrix (GLCM) features, which characterize texture based on pixel pairs with specific grey levels, and shape features (circularity and aspect ratio).

The feature details were used to train a random forest model with the SKLearn RandomForestClassifier. The initial model had an accuracy score of 0.569 and an F1 score of 0.57. Tested on the validation set, the model resulted in the confusion Matrix in Figure 1.

Class	Precision	Recall	F1-score
Non-toxic (0)	0.57	0.58	0.57
Toxic (1)	0.57	0.56	0.57
Accuracy			0.57

TABLE II: Performance Metrics for Random Forest

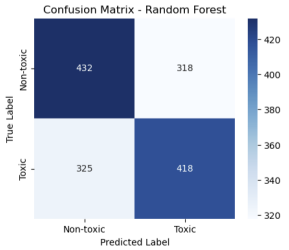

FIG. 1: Random Forest Confusion Matrix

Then, a random forest model was trained with tuned hyperparamters, as determined by a grid search. The tuned model's accuracy improved to 0.6, but the increase was not significantly better than chance, suggesting that a deep-learning approach might be beneficial.

Class	Precision	Recall	F1-score
Non-toxic (0)	0.61	0.57	0.59
Toxic (1)	0.59	0.62	0.61
Accuracy			0.60

TABLE III: Performance Metrics for Random Forest Tuned

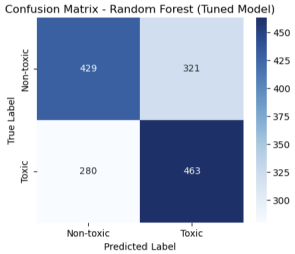

FIG. 2: Random Forest Tuned Confusion Matrix

B. Convolutional Neural Network Classification

In order to improve upon the accuracy of the random forest model, variations of a CNN model were trained and tested to classify toxic and non-toxic plants. An EfficientNetB0 model was chosen because for its ability to balance accuracy with computational complexity and was trained with various hyperparameter weights and Convolutional Block Attention Modules (CBAM). The implementation of CBAM can be used to test the SCAM-Herb model developed by Azadnia et al., which incorporates spatial attention and channel attention modules. [1]

Model Variation	Training	Validation	Test
Random Weights, L Aug	0.5062	0.4950	0.511721
Pretrained Weights, L Aug	0.6730	0.6611	0.659076
CBAM, Rand Weights, L Aug	0.6110	0.5928	0.585399
CBAM, Pretrained Weight, L Aug	0.9937	0.8292	0.815807

TABLE IV: Model Variation Accuracy on Training/Validation/Test Datasets

EfficientNetB0 Model Variation	Accuracy	Precision
Random Weights, Light Aug	0.511721	0.529412
Pretrained Weights, Light Aug	0.659076	0.653061
CBAM, Random Weights, Light Aug	0.585399	0.587601
CBAM, Pretrained Weights, Light Aug	0.815807	0.820270

TABLE V: Model Variation Test Accuracy & Precision

First, an EfficientNetB0 model with random weights was trained, with the hyperparameters of batch_size=32, num_epochs = 10 learning_rate = 1e-4. This model was trained and validated, achieving a training accuracy of 50.62% and a validation accuracy of 49.50%. It had a test accuracy of 51.17%. These metrics were worse than the tuned random forest model and about equal to chance.

Then an EfficientNetB0 model with random weights and CBAM was trained, with the same hyperparameters and sequential channel attention and spatial attention. This model was trained and validated, achieving a training accuracy of 61.10% and a validation accuracy of 59.28%. It had a test

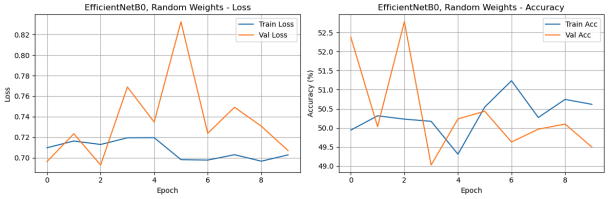

FIG. 3: EfficientNetB0 - Random Weights Loss and Accuracy

accuracy of 58.53%. These metrics were slightly better than the model without CBAM, but slightly worse than the tuned random forest model and still about equal to chance.

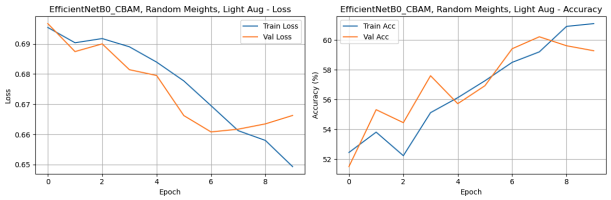

FIG. 4: EfficientNetB0 CBAM - Random Weights Loss and Accuracy

Next an EfficientNetB0 model with pretrained weights was trained. This model was trained and validated, achieving a training accuracy of 67.30% and a validation accuracy of 66.11%. It had a test accuracy of 65.09%. These metrics were better than the tuned random forest model and the random weights EfficientNetB0 model.

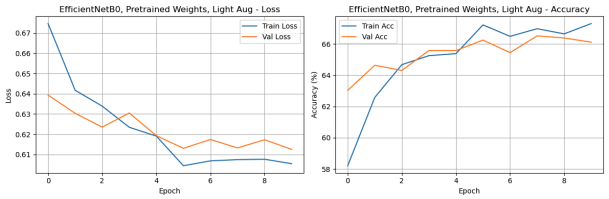

FIG. 5: EfficientNetB0 - Pretrained Weights Loss and Accuracy

Then an EfficientNetB0 model with CBAM was trained that also included pretrained weights to refine feature maps. This model was trained and validated, achieving the highest training accuracy of 99.37% and a validation accuracy of 82.92%. It had by far the highest test accuracy of 81.58%. These metrics were better than all other models tested and reflect a model that could reasonably be able to tell whether a plant is toxic or not with a fair degree of certainty. The better accuracy scores for this model indicate that the channel attention module and spatial attention module are supporting better results by focusing on the most relevant channels and spatial locations.

V. Next Steps

The next steps in this study are threefold. First, find further ablation methods to increase the accuracy of the model and/or tweak the current ablations to seek improvement. Due to the diversity of backgrounds found in the iNaturalist dataset, further investigation into data augmentation could improve model accuracy. Adding additional transformations on the

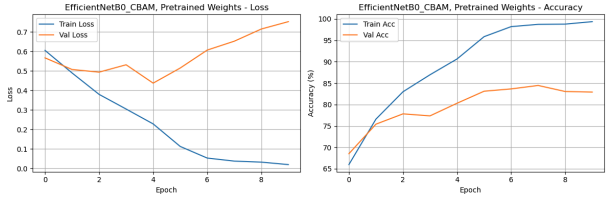

FIG. 6: EfficientNetB0 CBAM - Pretrained Weights Loss and Accuracy

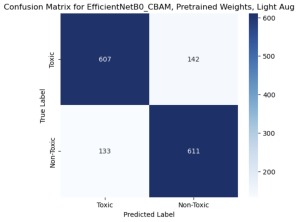

FIG. 7: EfficientNetB0 CBAM - Confusion Matrix

training dataset could improve the model's ability to classify images with a variety of backgrounds and lighting. Second, we have only experimented with EfficientNet-B0. There are other EfficientNet models (B1-B7). While the bigger models may tend to overfit to our data, it is worth exploring to see if a more powerful model can achieve higher accuracy before overfitting becomes a problem. If these steps are achieved and we find we have extra time, we may revisit an option to classify plant species, but that may be for another study. Finally, the model was trained using 10 epochs; however, increasing the number of epochs may yield higher accuracy. The optimal number of epochs for improving accuracy without overfitting will be explored in the next steps.

VI. Member Contributions

All group members have contributed to this project. Margaret and Brian wrote code to read in the data and build the EfficientNetB0 model as well as the CBAM module. Katherine worked on the Random Forest model and capturing metrics for all models, including the confusion matrices. Katherine also synthesized results and formatted data in Overleaf, and all group members contributed to updating the report.

References

[1] Azadnia, Rahim, Faramarz Noei-Khodabadi, Azad Moloudzadeh, Ahmad Jahanbakhshi, and Mahmoud Omid. "Medicinal and Poisonous Plants Classification from Visual Characteristics of Leaves Using Computer Vision and Deep Neural Networks." Ecological Informatics 82 (September 2024): 102683. https://doi.org/10.1016/j.ecoinf.2024.102683.

[2] Hridoy, Rashidul Hasan, Fatema Akter, and Maisha Afroz. "An Efficient Computer Vision Approach for Rapid Recognition of Poisonous Plants by Classifying Leaf Images Using Transfer Learning." 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), IEEE, July 6, 2021, 01–07. https://doi.org/10.1109/ICCCNT51525.2021.9580011.

[3] Cheek, Martin, Eimear Nic Lughadha, Paul Kirk, et al. "New Scientific Discoveries: Plants and Fungi." Plants, People, Planet 2, no. 5 (2020): 371–88. https://doi.org/10.1002/ppp3.10148.

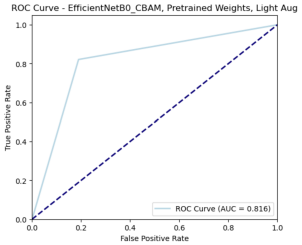

FIG. 8: EfficientNetB0 CBAM - ROC Curve

[4] Zuhri, Mohammad Faishol, S. Kholidah Rahayu Maharani, Affandy, Aris Nurhindarto, Abdul Syukur, and Moch Arief Soeleman. "Classification of Toxic Plants on Leaf Patterns Using Gray Level Co-Occurrence Matrix (GLCM) with Neural Network Method." Journal of Development Research 6, no. 1 (2022): 1–5. https://doi.org/10.28926/jdr.v6i1.202.

[5] Noor, T.H.; Noor, A.; Elmezain, M. Poisonous Plants Species Prediction Using a Convolutional Neural Network and Support Vector Machine Hybrid Model. Electronics 2022, 11, 3690. https://doi.org/10.3390/electronics11223690

[6] Marwa, Muskaan, P. Aman Reddy, M. Nithish Reddy, and M. Satyanarayana Reddy. "Toxic Plant Classification Using Convolutional Neural Networks." REST Journal on Data Analytics and Artificial Intelligence 4, no. 1 (2025): 192–200. https://doi.org/10.46632/jdaai/4/1/23.

[7] Kolhar, Shrikrishna, and Jayant Jagtap. "Plant Trait Estimation and Classification Studies in Plant Phenotyping Using Machine Vision – A Review." Information Processing in Agriculture 10, no. 1 (2023): 114–35. https://doi.org/10.1016/j.inpa.2021.02.006.

[8] Elliot, Hans. "Toxic Plant Classification." 2022. https://www.kaggle.com/datasets/hanselliott/toxic-plant-classification/data

[9] Ghysels, S.; De Baets, B.; Reheul, D.; Maenhout, S. (2025) Image-based yield prediction for tall fescue using random forests and convolutional neural networks. *Front. Plant Sci.* 16:1549099. https://doi.org/10.3389/fpls.2025.1549099