Improving Rare Tree Species Classification using Domain Knowledge

Ira Harmon, Sergio Marconi, Ben Weinstein, Yang Bai, Daisy Zhe Wang, Ethan White, Stephanie Bohlman

Abstract

Forest inventory forms the foundation of forest management. Remote sensing (RS) is an efficient means of measuring forest parameters at scale. Remotely sensed species classification can be used to estimate species abundances, distributions, and to better approximate metrics such as above ground biomass. State of the art methods of RS species classification rely on deep learning models such as convolutional neural networks (CNN). These models have 2 major drawbacks: they require large samples of each species to classify well and they lack explainablity. Therefore, rare species are poorly classified causing poor approximations of their associated parameters. We show that the classification of rare species can be improved by as much as 8 F1-points using a neuro-symbolic (NS) approach that combines CNNs with a NS framework. The framework allows for the incorporation of domain knowledge into the model through the use of mathematically represented rules, improving model explainability.

Index Terms

neuro-symbolics, explainable machine learning, remote sensing, tree species classification, convolutional neural network.

I. Introduction

Forests play a vital role in maintaining life on Earth. They store carbon, are a habitat for countless animals and provide fuel and production materials for numerous industries. As a result, governments and the forestry industry invest heavily in forest monitoring and management. Traditional inventory methods rely on manual field surveys that are used to estimate forest parameters such as biomass, tree mortality rates, species abundances, and species distributions based on sampling plots within the forest [1]. Though standard field survey plots are 1 hectare or less in area, manual sampling is labor intensive and therefore the number of plots inventoried

is limited by available people-power. Limited sampling ability hampers high precision estimates of forest parameters at scale.

Since the 1970's, remote sensed data products have become readily available [2]. RS data products can include optical images such as RGB and hyperspectral (HS), as well as LiDAR point clouds and synthetic aperture radar (SAR) returns. With the help of automation, these data products are used for forest monitoring at scales of 10's to 1000's of hectares [3], [4].

Recognizing species from RS data products is termed species classification. Accurate species classification is particularly important for measuring species abundances, species distributions, and biodiversity; non-species-specific metrics such as above ground biomass and basal area may be estimated more accurately when species is taken into account [5]. Methods for classifying species based on LiDAR, HS images, RGB images, SAR returns and almost every combination of the aforementioned modalities have been developed [6]. Here we focus on optical imagery.

Early methods of species classification used parametric statistical models such as linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) or methods like maximum likelihood estimation (MLE) [6]. Most modern methods use decision tree based classifiers or neural models on LiDAR, RGB, or HS data [6]–[8], with studies suggesting that using HS data gives superior performance. Advanced deep models for species classification include CNN's (2D and 3D), CNN's with attention, and transformers [9], [10].

Neural models have several drawbacks. Most prominently, they typically require large training sets, can be computationally intensive to train, and have low explainability [11], [12]. Guidelines for training deep neural models suggest 1,000's of instances of each class for optimal performance [12], [13]. Unfortunately, datasets are built by sampling from the real world and ecological systems like forests typically contain a few common species and many rare species [14]. Trees that are rare in a forest are likely to be rare within the dataset. This means that neural models for species classification are typically poor at recognizing rare species. Depending on the application, a species' frequency within the dataset may or may not positively correlate with the importance of its recognition to the user. An analysis of challenges inherent in rare species classification can be found in [15].

One approach to reducing dataset size requirements and improving explainability is neuro-symbolics (NS). NS architectures are a combination of neural and symbolic models [16]. Symbolic models use logical formalisms or distance metrics to make inferences. Domain knowledge in symbolic models is usually represented as a rule, an equation, a knowledge base, or knowledge

graph. First order logic (FOL) and propositional logic are two commonly used formalisms for creating models where inferences are made by reasoning over dataset instances with a set of rules [17], [18].

While neural models are good at learning from labeled examples, the "reasoning" behind their inferences is generally unclear to humans. By comparison, models that make inferences based on symbolic representations of data tend to have higher explainability, but may learn poorly from examples. The idea behind NS is that by combining the two approaches, we can capture the best of both worlds: the high explainability of symbolic models with the learning capacity of neural models. Studies have also shown that NS models are better able to learn in data constrained settings compared to purely neural models [19]. In this work, we leverage this property to improve classification on rare species.

The use of NS models for species classification is not new to ecology. [20] combines a convolutional neural network (CNN) with a knowledge graph and text embeddings to classify bird species from RGB images. [21] combines a CNN with text embeddings to classify tree species from RGB images. However, the frameworks and methods used by [20] and [21] are not easily applied to other models and require the user to find auxiliary data in the form of text or knowledge graphs to embed for semantic reasoning.

To address these shortcomings we propose using a modified version of DeepCTRL, a NS framework created by Google that uses a form of semantic regularization [22]. DeepCTRL allows the user to create rules as equations that incorporate domain knowledge into a neural network through its loss function. By incorporating a rule as a term in the loss function, the model is penalized during optimization for both incorrect inferences and inferences that break rules. Therefore during training, optimum performance occurs when correct inferences are made without breaking the rule. Because the model is forced to follow a known rule, and the degree to which a rule is followed can be estimated from the training loss, the model becomes more explainable. Ideally, the model would be able to learn the rule solely from the training data, but due to noise and other factors, this is not always the case. Our method gives a simple way for users to build NS and thus explainability into their models.

II. Data

The dataset for our study comes from the Tea Kettle Experimental Forest (TEAK). TEAK is one of 81 sites monitored by the National Ecological Observatory Network (NEON). TEAK

is a mixed coniferous forest in the Sierra National Forest east of Fresno California at 36°58' N latitude and 119°1' W longitude. See [23] and [24] for a full description of its ecological characteristics.

NEON annually surveys monitored forests from an airborne observation platform that is instrumented with RGB and hyperspectral cameras and both discrete and full-waveform LiDAR. Flights occur annually over monitored sites when the ecosystem is in a period of peak greenness. The resolution of RGB and hyperspectral data products are 0.1 m and 1 m respectively [25].

The dataset we use was curated for [9]. It consists of HS and RGB rasters, along with a co-registered canopy height model (CHM). Data comes from a 2017 NEON survey of TEAK along with a field sample conducted by Fricker et al. in September of 2017. We supplement the dataset with a digital elevation model (DEM) for TEAK created by the US Geological Survey [26]–[28]. See [9] for more information on dataset curation.

The curated dataset has 8 classes, white fir (Abies concolor), red fir (Abies magnifica), incense cedar (Calocedrus decurrens), Jeffrey pine (Pinus jeffreyi), sugar pine (Pinus lambertiana), black oak, (Quercus kelloggii), lodgepole pine (Pinus contorta), and "dead". Standing dead trees of any species are assigned this label. Table I gives the number of trees in each class and its abbreviation.

Using the CHM and DEM we identified differences in the structural traits and topographic preferences of the species within this dataset to be used as the foundation for symbolic rules. The left plot in Fig. 1 shows the distribution of each species' height as represented by the dataset. At this site, black oak (quke) and lodgepole pine (pico) are shorter compared to other species in the dataset and distinct from each other in overall height distribution. Therefore we use maximum crown height from the training data as the foundation for a pair of rules demonstrating how to leverage the structural traits of species (Rules 1 and 2). The right plot in Fig. 1 gives the distribution of each species' elevation range within the dataset. A number of species show distinctive elevational distributions at the site. We chose the minimum elevation for red fir (abma) as the basis for a rule demonstrating how to leverage topographic distribution limits (Rule 3). Finally, we also demonstrate the use of a rule based only on the imagery itself to differentiate between living and dead trees using the green leaf index (GLI; Rule 4) [29].

TABLE I

THE NUMBER OF TREES AND PIXEL PATCHES IN EACH CLASS.

Code	Species	Abbreviation	Tree Count	Patch Count
0	white fir	abco	119	2,908
1	red fir	abma	47	851
2	incense cedar	cade	66	1,853
3	Jeffrey pine	pije	164	4,384
4	sugar pine	pila	68	2,740
5	black oak	quke	18	111
6	lodgepole pine	pico	62	895
7	any species	dead	169	3,520
	Total	**—**	**713**	**17,262**

III. METHODOLOGY

For classification we use the model from [9], an 8 layer fully-convolutional CNN. The model architecture is shown in Fig. 2.

We combine the Fricker CNN with the DeepCTRL framework. DeepCTRL is a model and data agnostic NS framework that is easy to use. The framework is composed of a task encoder, a rule encoder, and a decision block (see Fig. 2b). The loss function is a linear combination of task loss and rule loss, where task loss is the loss contributed by the model's failure to predict a label and the rule loss is contributed from the model's failure to follow a rule.

Following the protocol from [9] we create train, validation, and test sets. Using stratified sampling, the dataset is composed of 15 x 15 pixel patches sampled from the set of tree crowns. Again following the protocol in [9], we use 10 fold cross validation and report the mean of the macro-F1 score for each fold and the mean F1 score for the class on which each rule is based.

For our study we focus on RGB images. While it is possible to apply our approach to HS images, RGB imagery is much more widely available and the model we used made few mistakes

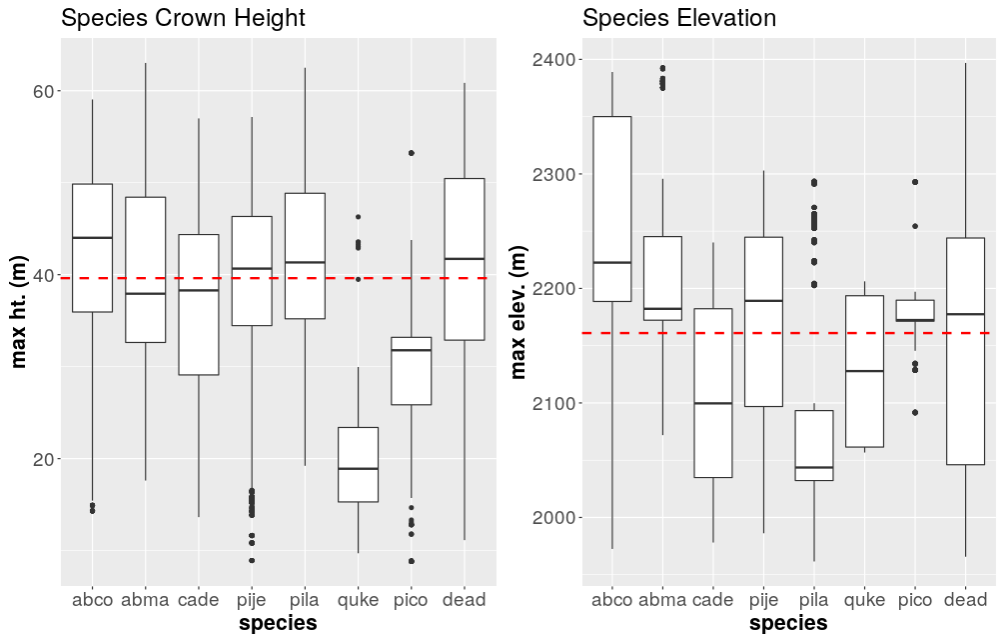

Fig. 1. The left plot shows distribution of species' crown height. The right plot shows distribution of species' elevation. The red lines indicate sample means.

that are correctable with domain knowledge when trained on HS images.

We use DeepCTRL as described in [30] with some modifications. Because DeepCTRL is data agnostic it can be made to work with any type of input. In our case, the input is a 15 x 15 patch of an RGB image created from the aforementioned NEON geotiffs. We concatenate the image with auxiliary data, a 15 x 15 patch of a co-registered CHM or DEM raster. In the case of the DEM, the raster is scaled by one-tenth so its values are of the same order as the values of the RGB geotiff.

After removing the final output stage, we use the CNN from the Fricker model as both the task and rule encoder. $\mathbf{z_d}$ and $\mathbf{z_r}$ are the output of convolution layer 5 (shown in Fig. 2a) from the task and rule encoder respectively. The decision block is composed of a convolutional layer with an input dimension of 256 and an output dimension of 8. Finally, the output of the decision block is passed through a softmax layer.

In the original design, during training, $\mathbf{z_d}$ and $\mathbf{z_r}$ are scaled by the constants α and $1-\alpha$. α is sampled from a β-distribution. This allows the model to learn varying degrees of rule enforcement during training. At inference, the user can vary the value of α depending on the strength of their belief in how much the rule is followed in the test set. We obtained better results

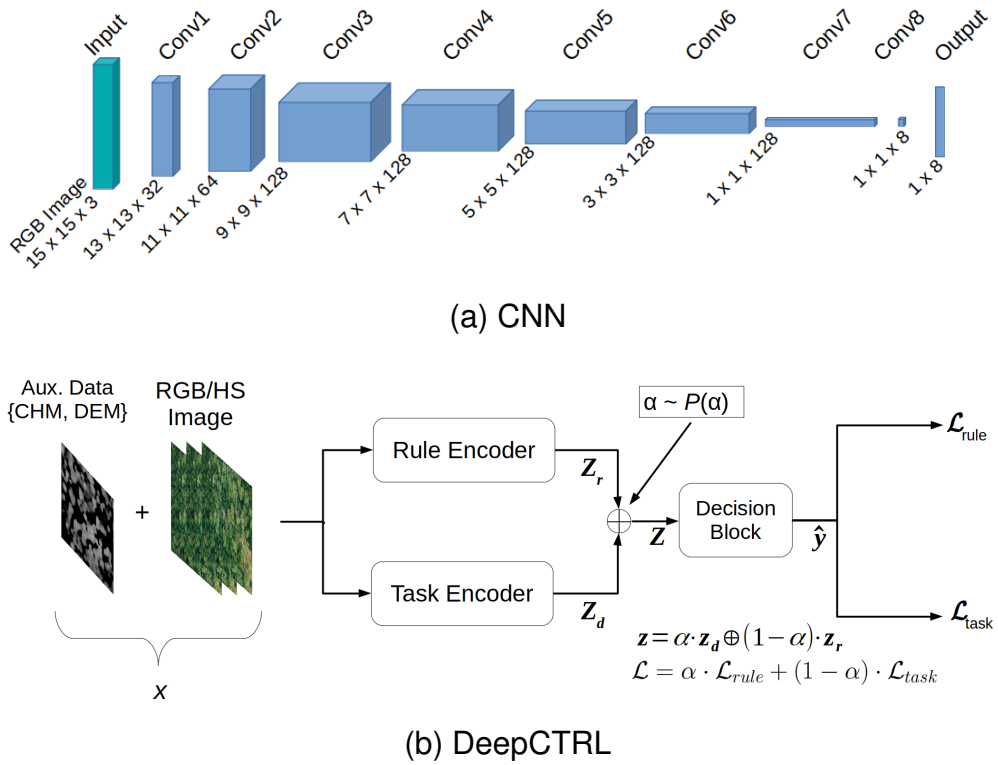

Fig. 2. CNN and DeepCTRL architectures.

by fixing α at 0.4 for both training and inference. By fixing α, the model loses its ability to alter how strongly the rule is adhered to post training, but gains in performance. A pseudocode description of the algorithm is given in [30].

We use the following notation. Dataset $\mathcal{D}$ consists of tuples of inputs from set $\mathcal{X}$ and labels from set $\mathcal{Y}$ where $\mathcal{X}$ is the set of pixel patches and $\mathcal{Y}$ is the set of their species labels: $\mathcal{D} = \{(\mathbf{x_1}, \mathbf{y_1}), (\mathbf{x_2}, \mathbf{y_2}), ..., (\mathbf{x_n}, \mathbf{y_n})\}$. Each label $\mathbf{y_i}$ is a an 8-way 1-hot encoding. Each model prediction, $\hat{\mathbf{y}}$, is an 8-way probability simplex.

The loss function is composed of the linear combination of two simpler loss functions, $\mathcal{L}_{rule}$ and $\mathcal{L}_{task}$. $\mathcal{L}_{task}$ is the cross entropy loss between $\mathbf{y}$ and $\hat{\mathbf{y}}$ as shown in (1).

$$\mathcal{L}_{CE}(\mathbf{y}, \hat{\mathbf{y}}) \tag{1}$$

We define $\mathcal{L}_{rule}$ as the cross-entropy loss between a function, ϕ, and $\hat{\mathbf{y}}$ where $\mathbf{x}$ is a training instance and

$$\phi : \mathbf{x} \rightarrow \ \ u \in (0, 1). \tag{2}$$

$\mathcal{L}_{rule}$ then becomes

$$\mathcal{L}_{CE}(\phi(\mathbf{x}), \hat{\mathbf{y}}_k) \cdot \mathbf{1}(\hat{\mathbf{y}}_k = +) \tag{3}$$

where $\mathbf{1}(\cdot)$ is an indicator function, $\hat{\mathbf{y}}_k$ is k-th element of $\hat{\mathbf{y}}$, and the $+$ indicates the k-th class is predicted.

We define ϕ as the composition of two functions. An inner function, f where

$$f : \mathbf{x} \rightarrow t \in \mathbb{R}. \tag{4}$$

Function f quantizes how much $\mathbf{x}$ is in compliance with its respective rule. σ is a differentiable function that maps $f(\mathbf{x})$ to a value $\in [0, 1]$. We use the sigmoid function:

$$\sigma(t) = \frac{1}{1 + exp(-t)}. \tag{5}$$

For each rule, there is a threshold that we represent as a translation of the sigmoid along the x-axis. Depending on the domain knowledge, the presence or absence of the species of interest may only occur above or below the threshold. The function used for f varies with each rule. ϕ then becomes the composition of σ and f,

$$\phi = \sigma \circ f. \tag{6}$$

For rules 1-3 the internal function, f, takes the maximum value of the auxiliary data layer. For rule 4, which uses no auxiliary data, f calculates the GLI of $\mathbf{x}$ as

$$gli(\mathbf{x}) = \frac{2 \cdot G - R - B}{2 \cdot G + R + B} \tag{7}$$

where R, G, B are the pixel values in each RGB channel. The equations for each rule are given in the next section.

Rules 1 - 3 come from examining presence - absence cut-offs in the CHM and DEM distributions. Rule 4 comes from examining errors in the validation set confusion matrix referenced against GLI.

IV. Experiments

A. Experiment Setup

Following the protocol from [9], stratified sampling was used to create 10 folds of 15 x 15 pixel patches from the RGB, CHM, and DEM rasters. We created 4 rules. In natural language,

rule 1 states that if the height of a tree crown is over 46 m it is unlikely to be a black oak. We write this mathematically as

$$\phi_1(\mathbf{x}) = \frac{1}{1 + exp(-(1 \times 10^3(-\max_{CHM}(\mathbf{x}) + 46.0)))}. \tag{8}$$

Rule 2 states that trees taller than 53.2 m are unlikely to be lodgepole pine. We write rule 2 mathematically as

$$\phi_2(\mathbf{x}) = \frac{1}{1 + exp(-(1 \times 10^3(-\max_{CHM}(\mathbf{x}) + 53.2)))}. \tag{9}$$

Rule 3 states that trees growing at an elevation less than 2072 m are unlikely to be red fir. Rule 3 is written mathematically as

$$\phi_3(\mathbf{x}) = \frac{1}{1 + exp(-(-(1 \times 10^3(-\max_{DEM}(\mathbf{x}) + 2072))))}. \tag{10}$$

Rule 4 states that trees with a GLI less than 0.1 are unlikely to be incense cedar. Rule 4 is written mathematically as

$$\phi_4(\mathbf{x}) = \frac{1}{1 + exp(-(-(1 \times 10^3(-gli(\mathbf{x}) + 0.1))))}. \tag{11}$$

For rules 1 and 2 the RGB raster is augmented with the CHM by adding the CHM as a 4th channel. Similarly, for rule 3 the DEM is added as the 4th channel. These channels are also available to the baseline neural model when making comparisons. We use the patch classifier from [9] trained on the RGB image with auxiliary data as a baseline. Both baseline and experiment models are trained for 5 epochs using the Adam optimizer with L2 regularization and a learning rate of 1×10^{-4}. Finally, we perform an ablation study to determine how much each rule contributes to the change in model performance. For each rule we set a random threshold value for the CHM, DEM, or GLI between the minimum and maximum values present in the training dataset. The randomized values are selected from a uniform distribution. We repeat the ablation study 30 times for each rule and average the results as the difference between the experimental model with the threshold used in its respective rule and the experimental model with the randomized threshold.

B. Results and Analysis

Compared to the baseline, the rules had a mostly positive effect on performance. Fig. 3 shows that rules 1 and 2 improved both the overall F1 and the rule's class F1, while rule 3 caused

a reduction in the overall F1 but still improved its class F1. Differences are quantified as F1-points, where a 0.01 change in F1 is a change of 1 F1-point. Rule 1 improved overall F1 by 0.63 F1-points. The rule's class F1 was improved by 8.3 F1-points. For rule 2 overall F1 and class F1 improved by 0.43 and 1.84 F1-points respectively. Rule 3 worsened the F1 by 0.97 F1-points, but still increased class F1 by 0.6 F1-points. Rule 4 improved F1 by 0.59 F1-points and class F1 by 1.1 F1-points.

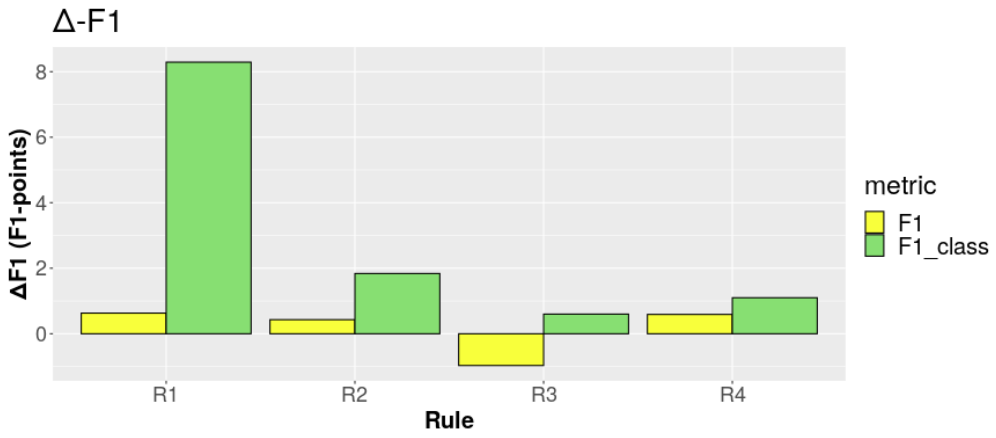

Fig. 3. The change in macro-F1 and the class specific F1 for each rule. Rule 1 had the biggest impact on performance.

Fig. 4 shows the changes in the confusion matrices between the baseline model and the experiment models for each rule. The recall columns are normalized by row and the precision columns are normalized by column. For rule 1 both precision and recall are improved. For class 5, black oaks, the precision is improved by 3 points and the recall by 12 points. The rule has the largest negative impact on the precision of class 1, which is reduced by 5 points.

Rule 2, which was designed to affect class 6, improves both precision and recall. Class precision improves by 2 points and class recall by 5 points. Rule 2 has the largest negative impact on the precision of class 1, reducing it by 4 points. Rule 2 also has a positive effect on class 5, improving its recall by 11 points.

Rule 3, which was written around class 1, improves class 1's precision and recall by 1 and 3 points respectively. It has a negative impact on the precision and recall of class 5. This is contrary to rules 1 and 2 which both improve class 5.

Rule 4, which is designed around class 2, improves class precision by 1 point. The overall F1 is improved by 0.59 points, while class F1 improves by 1.1 F1-points. This rule improves on the precision of class 5 by 2 points, while reducing class 5's recall by 3 points.

The rarest species was most affected by the inclusion of domain knowledge. We hypothesize that this effect is most profound when rules derived from domain knowledge are applicable to the dataset, but the model, due to noise, data imbalance, or other reasons, is unable to learn the rule from the data alone.

By rarity, species are ordered 5, 1, 6, 2, but by base model ascending class F1 performance they are ordered 1, 5, 6, 2. Rule 3, which affects class 1, had the largest ratio of the number of rule-correctable incorrect predictions to the number of total predictions, while rule 1 had the 2nd largest. Rule 3 which is designed for the 2nd rarest species with the worst base model performance is significantly less effective than rule 1, suggesting that the domain knowledge derived from rule 1 may be a better differentiator between species than the domain knowledge applied to rule 3.

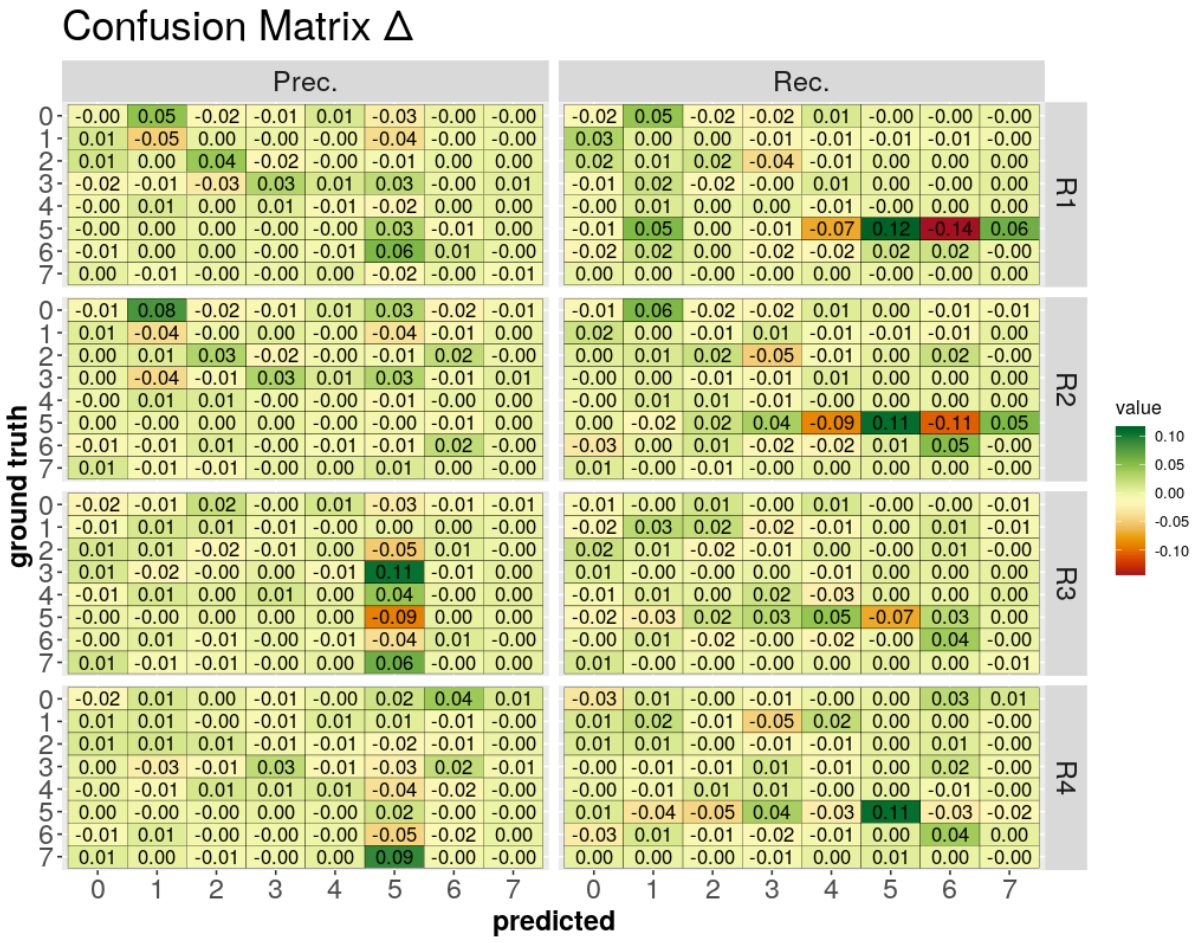

Fig. 4. The change in the confusion matrices for baseline and experiment models normalized by column for precision and row for recall.

The results of the ablation study are shown in Fig. 5. The results suggest that the influence of domain knowledge is strongest for rule 1, which is likely due to the rarity of black oak in the dataset. Nevertheless, each species for which a rule was created was impacted by the inclusion of domain knowledge. As in [30] the study suggests that there is a slight boost in performance when the model is placed in a NS framework and that this boost is independent of additional domain knowledge.

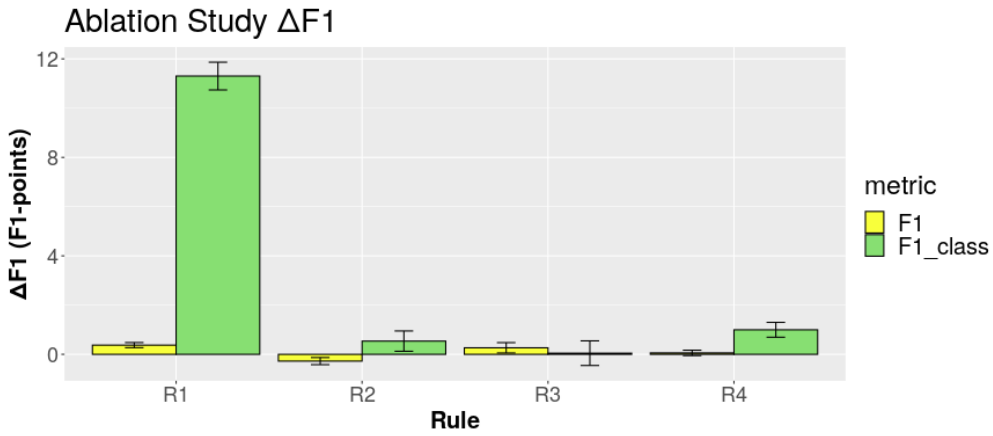

Fig. 5. The average difference between F1 and F1_class when using correct thresholds versus randomized thresholds. Error bars show 95% confidence intervals ($\mu \pm CI$; n=30).

V. Conclusion

In this work we show that domain knowledge can be encoded through a function and then injected into a species classification neural network. This method is more accessible than other NS frameworks that use formalisms such as FOL, knowledge bases, or text embeddings. Our results show that model performance on rare species can be significantly improved through the inclusion of domain knowledge using our method, which simply applies a slight modification to the original model architecture and adds an additional term to the loss function.

Acknowledgments

We would like to thank NEON. This work was made possible by NSF grant 1926542.

References

[1] W. A. Bechtold and P. L. Patterson, *The enhanced forest inventory and analysis program–national sampling design and estimation procedures*. USDA Forest Service, Southern Research Station, 2005, no. 80.

[2] D. Boyd and F. Danson, "Satellite remote sensing of forest resources: three decades of research development," *Progress in Physical Geography*, vol. 29, no. 1, pp. 1–26, 2005.

[3] G. P. Asner and R. E. Martin, "Airborne spectranomics: mapping canopy chemical and taxonomic diversity in tropical forests," *Frontiers in Ecology and the Environment*, vol. 7, no. 5, pp. 269–276, 2009.

[4] M. A. Wulder, J. C. White, R. F. Nelson, E. Næsset, H. O. Ørka, N. C. Coops, T. Hilker, C. W. Bater, and T. Gobakken, "Lidar sampling for large-area forest characterization: A review," *Remote sensing of environment*, vol. 121, pp. 196–209, 2012.

[5] F. E. Fassnacht, D. Mangold, J. Schäfer, M. Immitzer, T. Kattenborn, B. Koch, and H. Latifi, "Estimating stand density, biomass and tree species from very high resolution stereo-imagery–towards an all-in-one sensor for forestry applications?" *Forestry: An International Journal of Forest Research*, vol. 90, no. 5, pp. 613–631, 2017.

[6] F. E. Fassnacht, H. Latifi, K. Stereńczak, A. Modzelewska, M. Lefsky, L. T. Waser, C. Straub, and A. Ghosh, "Review of studies on tree species classification from remotely sensed data," *Remote Sensing of Environment*, vol. 186, pp. 64–87, 2016.

[7] M. Immitzer, C. Atzberger, and T. Koukal, "Tree species classification with random forest using very high spatial resolution 8-band worldview-2 satellite data," *Remote sensing*, vol. 4, no. 9, pp. 2661–2693, 2012.

[8] C. Zhang, K. Xia, H. Feng, Y. Yang, and X. Du, "Tree species classification using deep learning and rgb optical images obtained by an unmanned aerial vehicle," *Journal of Forestry Research*, vol. 32, no. 5, pp. 1879–1888, 2021.

[9] G. A. Fricker, J. D. Ventura, J. A. Wolf, M. P. North, F. W. Davis, and J. Franklin, "A convolutional neural network classifier identifies tree species in mixed-conifer forest from hyperspectral imagery," *Remote Sensing*, vol. 11, no. 19, p. 2326, 2019.

[10] P. Sun, X. Yuan, and D. Li, "Classification of individual tree species using uav lidar based on transformer," *Forests*, vol. 14, no. 3, p. 484, 2023.

[11] F. Xu, H. Uszkoreit, Y. Du, W. Fan, D. Zhao, and J. Zhu, "Explainable ai: A brief survey on history, research areas, approaches and challenges," in *CCF international conference on natural language processing and Chinese computing*. Springer, 2019, pp. 563–574.

[12] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 843–852.

[13] M. Huh, P. Agrawal, and A. A. Efros, "What makes imagenet good for transfer learning?" *arXiv preprint arXiv:1608.08614*, 2016.

[14] B. J. McGill, R. S. Etienne, J. S. Gray, D. Alonso, M. J. Anderson, H. K. Benecha, M. Dornelas, B. J. Enquist, J. L. Green, F. He *et al.*, "Species abundance distributions: moving beyond single prediction theories to integration within an ecological framework," *Ecology letters*, vol. 10, no. 10, pp. 995–1015, 2007.

[15] B. Weinstein, S. Marconi, S. Graves, A. Zare, A. Singh, S. Bohlman, L. Magee, D. Johnson, P. Townsend, and E. White, "Capturing long-tailed individual tree diversity using an airborne multi-temporal hierarchical model," *bioRxiv*, pp. 2022–12, 2022.

[16] A. S. Garcez, L. C. Lamb, and D. M. Gabbay, *Neural-symbolic cognitive reasoning*. Springer Science & Business Media, 2008.

[17] J. R. Quinlan, "Learning logical definitions from relations," *Machine learning*, vol. 5, no. 3, pp. 239–266, 1990.

[18] P. Clark and T. Niblett, "The cn2 induction algorithm," *Machine learning*, vol. 3, no. 4, pp. 261–283, 1989.

[19] Z. Hu, X. Ma, Z. Liu, E. Hovy, and E. Xing, "Harnessing deep neural networks with logic rules," *arXiv preprint arXiv:1603.06318*, 2016.

[20] H. Xu, G. Qi, J. Li, M. Wang, K. Xu, and H. Gao, "Fine-grained image classification by visual-semantic embedding." in *IJCAI*, 2018, pp. 1043–1049.

[21] G. Sumbul, R. G. Cinbis, and S. Aksoy, "Fine-grained object recognition and zero-shot learning in remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, no. 2, pp. 770–779, 2017.

[22] M. Diligenti, M. Gori, and C. Sacca, "Semantic-based regularization for learning and inference," *Artificial Intelligence*, vol. 244, pp. 143–165, 2017.

[23] "Lower Teakettle NEON/TEAK," 2019. [Online]. Available: https://www.neonscience.org/field-sites/teak

[24] "Teakettle Experimental Forest." [Online]. Available: https://www.fs.fed.us/psw/ef/teakettle/

[25] T. U. Kampe, B. R. Johnson, M. A. Kuester, and M. Keller, "Neon: the first continental-scale ecological observatory with airborne remote sensing of vegetation canopy biochemistry and structure," *Journal of Applied Remote Sensing*, vol. 4, no. 1, p. 043510, 2010.

[26] "USGS arc-second n37w119 1x1 degree: US Geological Survey," Tech. Rep., Jan. 2013. [Online]. Available: https://apps.nationalmap.gov/downloader/#/

[27] "USGS arc-second n38w119 1x1 degree: US Geological Survey," Tech. Rep., Jun. 2018. [Online]. Available: https://apps.nationalmap.gov/downloader/#/

[28] "USGS arc-second n37w120 and n38w120 1x1 degree: US Geological Survey," Tech. Rep., Mar. 2019. [Online]. Available: https://apps.nationalmap.gov/downloader/#/

[29] M. Louhaichi, M. M. Borman, and D. E. Johnson, "Spatially located platform and aerial photography for documentation of grazing impacts on wheat," *Geocarto International*, vol. 16, no. 1, pp. 65–70, 2001.

[30] S. Seo, S. Arik, J. Yoon, X. Zhang, K. Sohn, and T. Pfister, "Controlling neural networks with rule representations," *Advances in Neural Information Processing Systems*, vol. 34, pp. 11 196–11 207, 2021.