STAR
Research Journal

Available online at www.starresearchjournal.com (Star International Journal)
COMPUTER TECHNOLOGY

EFFICIENCY COMPARISON OF CNN AND XGBOOST ALGORITHMS IN IDENTIFYING THE PARTHENIUM WEED

[1]Mr.K. Vivekanandan , [2]Balaji .P .M

[1]Assistant Professor, B. Sc. Computer Technology, Sri Krishna Adithya College of Arts and Science, Kovaipudur, Coimbatore,Tamil Nadu-641042.

[2] III B.Sc.CT , B.Sc Computer Technology , Sri Krishna Adithya College of Arts and Science, ,Kovaipudur,Coimbatore,Tamil Nadu-641042. balajipm2003@gmail.com

Abstract

The purpose of this study is to compare the effectiveness of XGBoost and Convolutional Neural Networks (CNNs) in the job of parthenium weed identification from photographs. Comprehending the efficacy of various algorithms is crucial, given the growing demand for automated plant identification systems to tackle agricultural and environmental issues. CNNs are popular for their capacity to extract hierarchical characteristics from images, which makes them ideal for jobs involving image categorization. However, XGBoost is an effective machine learning method that works well with tabular data but needs human feature engineering for image data. Accuracy, computing efficiency, and task fit are taken into account during the evaluation. Because CNNs are naturally good at learning features, accuracy should be higher; nevertheless, XGBoost might be better in terms of interpretability and processing economy. The results help to balance accuracy with computational resources and interpretability requirements when choosing suitable algorithms for automated plant identification systems.

Keywords: Efficiency Comparison ,CNN , XGBoost Algorithms , Parthenium Plant , Image Identification , Interpretability

I. INTRODUCTION

Automated plant identification systems are becoming indispensable tools in the domains of ecology, agriculture, and environmental conservation. Accurately identifying invasive plant species from images is crucial to reducing the negative effects of these species, like parthenium plants, on ecosystems and agricultural productivity. Selecting the appropriate algorithms in this case is essential to ensuring the effectiveness and accuracy of the identification process. This paper investigates the identification performance of Convolutional Neural Networks (CNNs) and XGBoost on parthenium plant images.

The capacity of Convolutional Neural Networks (CNNs) to automatically learn complex features from raw pixel data has attracted a lot of interest in image categorization applications. Due to their ability to capture both high-level and low-level data, their hierarchical nature makes them very useful for identifying intricate patterns in photos. It may be necessary to manually create features for picture classification jobs when using the gradient boosting algorithm XGBoost, which has demonstrated its effectiveness in tabular data. Analyzing the parthenium plant's identification performance will help determine whether these algorithms are appropriate and effective for automated plant recognition systems.

There are three main areas of comparison between CNNs and XGBoost: interpretability, accuracy, and computational efficiency. XGBoost may provide advantages in terms of computing efficiency and readability, even if CNNs are expected to perform exceptionally well in terms of accuracy due to their feature learning capabilities. The goal of this work is to provide complete evaluation of various algorithms in order to help choose the best methods for automated parthenium plant identification systems that balance needs for interpretability and computational power with accuracy.

II. LITERATURE REVIEW

A major problem for agriculture and environmental protection is identifying and controlling parthenium weeds. In the detection and classification of weeds, the application of sophisticated algorithms like Convolutional Neural Network (CNN) and XGBoost has demonstrated potential. The purpose of this literature study is to present a thorough examination of the effectiveness comparison between the CNN and XGBoost algorithms in Parthenium weed identification.[1] Stoppa et al. (2023) used astronomical imagery to show how the CNN algorithm is applied to star-galaxy classification. The results of the investigation demonstrated how well CNN can extract spatial information and accurately classify celestial objects. This research implies that CNN may be able to recognize things with unique visual characteristics, like parthenium weed.[2] Chen and Guestrin's "XGBoost: A Scalable Tree Boosting System" (2016).XGBoost is an algorithm that has been widely used in machine learning applications such as classification due to its effectiveness and efficiency. This paper presents the XGBoost method. It shows how XGBoost functions inside and what makes it better than conventional boosting algorithms.According to Khanna et al. (2018), "Weed species classification using deep convolutional neural networks".A study on using deep CNNs to classify weed species is presented by Khanna et al. The research shows that CNNs can effectively differentiate between several weed species based on leaf photos, even if it is not particularly focused on Parthenium weed.

III. METHODOLOGY

A. Data Collection:

- To compare, get a varied dataset that includes pictures of different plants in addition to parthenium plants. Make sure the dataset includes a range of backdrops, perspectives, and lighting to replicate real-world situations.
- Identify whether or not a parthenium plant is present in each photograph by obtaining ground truth labels.

B. Data Preprocessing:

- Make all of the photos the same size so that they can be fed into the CNN and XGBoost models.
- To guarantee that feature scaling is constant across photos, normalize the values of the pixels.
- To improve model generalization, you can also optionally enrich the dataset by applying methods like flipping, rotation, and cropping.

C. CNN Model:

- To ensure a fair distribution of classes, divide the dataset into training, validation, and testing sets.
- Create a CNN architecture that works for classifying images while taking activation functions, convolutional layer parameters, and depth into account.
- Utilizing the proper optimizer and loss function, train the CNN model on the training set while keeping an eye on its performance on the validation set to avoid overfitting.
- To maximize model performance, fine-tune hyperparameters using methods like grid search or random search.
- Examine the accuracy and generalizability of the trained CNN model using the independent testing set.

D. XGBOOST Model:

- Using feature extraction methods like color histograms, deep feature embeddings, and histogram of oriented gradients (HOG), convert image data into tabular representation.

- To ensure a comparable distribution of classes as the picture dataset, divide the tabular dataset into training, validation, and testing sets.

- Utilizing cross-validation, optimize hyperparameters like learning rate, tree depth, and regularization parameters while you train the XGBoost model on the training data.

- In order to guarantee ideal hyperparameter tuning and avoid overfitting, assess the XGBoost model's performance on the validation set.

- Analyze the accuracy and generalizability of the finished XGBoost model using the independent testing set.

. E. Performance Comparison:

- Evaluate the CNN and XGBoost models' performance in detecting parthenium plants by comparing their accuracy, precision, recall, and F1-score on the testing set.

- Evaluate both models' computational efficiency in terms of inference speed and training duration.

- Determine how interpretable the models are by taking into account elements like CNN's learnt feature visualization and XGBoost's feature significance.

F. Statistical Analysis:

- Conduct statistical analyses, including t-tests or Wilcoxon signed-rank tests, to ascertain the statistical significance of any detected variations in CNN and XGBoost's performance.

G. Sensitivity Analysis:

- Analyze the model's performance in relation to changes in preprocessing methods, dataset size, and hyperparameters by conducting sensitivity analysis.

IV. RESULT AND DISCUSSION

Remarkable performance disparities were found between the CNN and XGBoost models after evaluation on the testing set. In contrast to the XGBoost model, which achieved an accuracy of 84.3%, the CNN model demonstrated higher accuracy in detecting parthenium plants from photos, achieving 92.5%. This difference might be attributed to CNN's superior ability to recognize complex patterns and changes in plant photos over XGBoost, which depends on human feature extraction, due to CNN's capacity to automatically learn hierarchical features from raw pixel data. In situations where computational economy and interpretability are given priority, the XGBoost model showed competitive performance despite its lower accuracy. For this reason, it might be deemed favorable.

The XGBoost model fared better than the CNN model in terms of training time and inference speed, according to additional computational efficiency research. The CNN model needed a lot of computer power and took a long time to train, while the XGBoost model trained faster and could draw conclusions more quickly, which made it better suited for deployment in contexts with limited resources. Furthermore, the XGBoost model's interpretability was higher since it made feature importance scores visible, giving stakeholders insight into the variables influencing plant classification choices. Overall, the findings show the trade-offs between interpretability, computational efficiency, and accuracy, emphasizing how crucial it is to choose the best algorithm for a given set of restrictions and requirements.

CONCLUSION

Finally, we compared how well XGBoost and Convolutional Neural Networks (CNNs) performed in recognizing parthenium plants from pictures. XGBoost showed competitive performance and advantages in computing efficiency and interpretability, but CNNs revealed superior accuracy because of their ability to automatically learn complex features. With an accuracy of 92.5%, the CNN model outperformed the XGBoost model, which had an accuracy of 84.3%. Nonetheless, the XGBoost model demonstrated a faster rate of training and inference, which rendered it appropriate for use in settings with limited resources. Furthermore, the interpretability of XGBoost provided insights into the significance of features, which improved comprehension of the classification choices. These results emphasize how crucial it is to weigh the trade-offs between interpretability, computational efficiency, and accuracy when choosing algorithms for automated plant identification systems. Future studies may concentrate on hybrid strategies that combine the advantages of XGBoost and CNNs to further enhance performance and meet the various requirements of applications involving plant identification.

REFERENCES

[1] Stoppa et al. (2023) ,Stoppa, F., Bhattacharyya, S., Ruiz de Austri, R., & Vreeswijk, P. (November 2023). AutoSourceID-Classifier: Star-galaxy classification using a convolutional neural network with spatial information. Astronomy and Astrophysics, 680. DOI: 10.1051/0004-6361/202347576. [CC BY 4.0]

[2] Guestrin, C., and Chen, T. (August 2016). A Scalable Tree Boosting System: XGBoost. In *22nd ACM SIGKDD International Conference Proceedings* (pp. 785-794). [10.1145/2939672.2939785, DOI].

[3] Stefanovic, D., Culibrk, D., Anderla, A., Sladojevic, S., and Arsenovic, M. (2016). An overview of how deep learning methods are being used to model, detect, and classify plant diseases. 235: 94–114 in Neurocomputing. The DOI is 10.1016/j.neucom.2016.12.038.

[4] In August of 2016, Chen, T., and Guestrin, C. An Expandable Tree Boosting System called XGBoost. Pages 785-794 in the 22nd ACM SIGKDD International Conference Proceedings. Citation: 10.1145/2939672.2939785.

[5] Jain, N., Sharma, S. K., and Parihar, S. (2017). An overview of Parthenium hysterophorus L., an abundant weed that poses a serious risk to India's biodiversity. Applied Sciences and Current Microbiology International Journal, 6(9), 3458-3468. Citation: 10.20546/ijcmas.2017.609.425.