

Research Paper



Optimizing tree-based algorithms for student dropout prediction: a comparative study

Roman B. Villones* 

*College of Informatics, Philippine Christian University, Metro Manila, Philippines.

Article Info

Article History:

Received: 03 April 2025

Revised: 10 June 2025

Accepted: 18 June 2025

Published: 06 August 2025

Keywords:

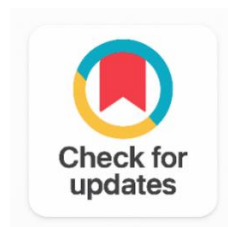
Dropout Prediction

Tree-Based Algorithms

Machine Learning

Optimization Techniques

Dragonfly Workflow



ABSTRACT

This study aims to predict student dropout by experimenting with different tree-based algorithms and optimization techniques. The goal is to improve prediction accuracy and support early intervention strategies in education. Applying the Dragonfly workflow, a structured and rigorous approach in machine learning. A range of tree-based algorithms including Decision Tree, Gradient Boosting, Random Forest, Extra Trees, Histogram-Based Gradient Boosting, Cat Boost, XG Boost, and Light GBM are evaluated. Optimizing its performance by hyperparameter tuning and cross-validation is conducted using Randomized Search CV, Grid Search CV, and Bayes Search CV. Among all algorithms, XG Boost consistently showed a high accuracy and slightly improved with BayesSearchCV. The Hist Gradient Boosting achieved the highest score overall under BayesSearchCV. It achieved the goal of building a robust predictive framework by combining EDA with tree-based algorithms. Key models were optimized and the Bayes Search CV showing the most consistent performance, particularly for XGBoost and HistGradient Boosting that highlighting the value of effective tuning for accurate student dropout prediction. Future work should begin with structured EDA and prioritize tree-based models for high performance. The BayesSearchCV is recommended as a standard approach to improve model effectiveness across domains.

Corresponding Author:

Roman B. Villones

College of Informatics, Philippine Christian University, Metro Manila, Philippines.

Email: roman.villones@pcu.edu.ph

Copyright © 2025 The Author(s). This is an open access article distributed under the Creative Commons Attribution License, (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

Student dropout is a persistent challenge in educational institutions worldwide that leads to significant academic, economic, and social consequences. With the growing availability of educational data and advances in computing, the machine learning (ML) has become an increasingly popular approach to predicting student attrition. Unlike traditional statistical models, machine learning can uncover complex, nonlinear relationships and generate predictive insights in real time [1].

At University of the Philippines Diliman highlighted that absenteeism, poor academic standing, and program mismatch are major predictors of early college dropout. They use predictive modeling that emphasizes the importance of pre-college academic history and course alignment [2]. Furthermore, the financial difficulties remain a primary driver of attrition, despite free tuition policies. They also identified mental health challenges like academic stress and family-related pressures as increasingly influential in the post-pandemic higher education landscape [3]. Dropouts in Cebu's tertiary institutions found that students often discontinue their studies due to economic burdens like personal responsibilities and a lack of academic engagement.

These findings reinforce that dropout is not solely an academic issue but one influenced by psychosocial and environmental pressures [4]. The objective of this study is to develop a robust tree-based predictive modeling framework by integrating exploratory data analysis, machine learning algorithms and optimization techniques. Initially, conduct a comprehensive exploratory data analysis for better understanding the patterns and how different parts of the data are connected. Subsequently, applying tree-based machine learning algorithms to construct a predictive models based on the insights derived from the exploratory data analysis. Furthermore, implementing a hyperparameter tuning for optimization and cross-validation techniques for better fitting of the model. Collectively, these objectives aim to contribute to the effective application of machine learning algorithms for accurate and reliable prediction in the student attrition.

2. RELATED WORK

Applying a C4.5 decision tree algorithm to secondary school data in Bangladesh and found that it could accurately classify at-risk students based on attendance, academic performance, and family background. While Decision Tree's are prone to overfitting, their clear rule-based outputs make them valuable for explaining decisions to educators [5]. Based on another study, they used decision tree visualizations to provide educators with understandable insights into dropout risk factors that facilitate the targeted interventions [6]. Another study presenting that Random Forest outperformed other models in predicting dropout in Indian engineering colleges, achieving over 90% accuracy. The study also emphasized the importance of feature selection, noting that semester-wise grades and attendance were the most influential predictors [7].

The Random Forest classifiers effectively identify at-risk students by analyzing academic records, attendance, and socio-demographic features [8]. Some studies used XG Boost to predict dropout likelihood in online university programs. The results showed a superior performance compared to logistic regression and support vector machines especially when using SH apley Additive ex Planations values to interpret feature contributions. The model identified prior academic performance, course engagement metrics, and socioeconomic factors as the most impactful features [9].

Applying XG Boost to a large dataset of university students and reported an accurate improvement of 10% compared to logistic regression models [10]. In addition, applying of Light GBM on data from a Brazilian higher education institution. The model delivered both high accuracy with quick response times, making it a good fit for dynamic learning environments. The findings support the deployment of automated alert systems that help educators intervene before a student drops out [11].

Applying optimization techniques like Grid Search CV to tune Random Forest classifiers for dropout prediction and reported a significant improvement in accuracy and F1-score compared to default

settings. However, Grid Search CV can be high in computational cost, especially when the hyperparameter space is large [12].

Utilizing Randomized Search CV to enhance the Gradient Boosting algorithm for dropout detection in high school students. Their approach reduced computation time by 60% while maintaining comparable predictive performance to Grid Search CV [13]. Demonstrated that Bayes Search CV outperformed both Randomized Search CV and Grid Search CV in tuning XG Boost models for university dropout prediction [14].

3. METHODOLOGY

This study implements the Dragonfly workflow [15] that provides a structured workflow and emphasizes cautious data selection, preprocessing, and partitioning by splitting the data into training, validation, and testing so it can prevent overfitting while improving the model's ability, as shown in Figure 1.

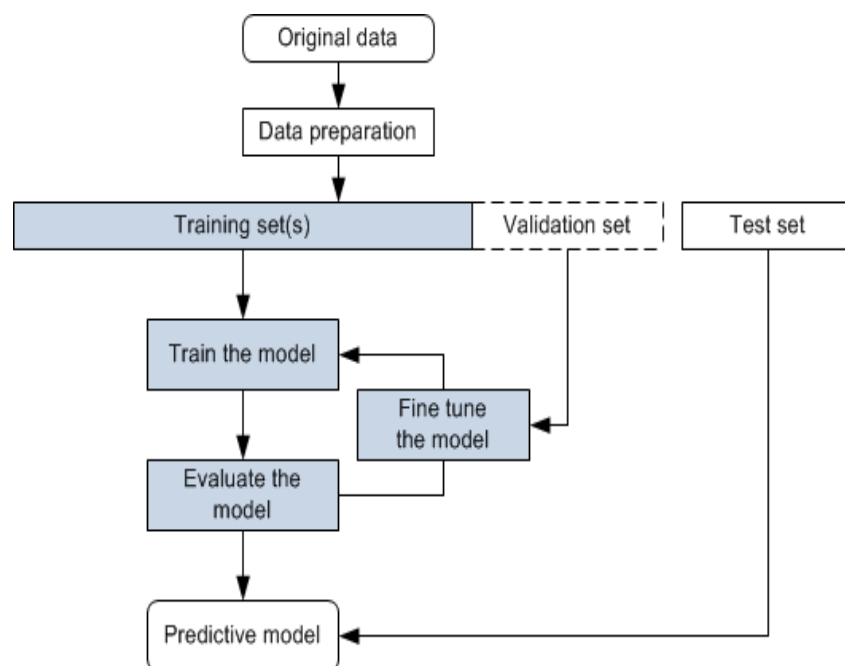


Figure 1. Dragonfly Workflow Pipeline

3.1. Original Data

The dataset utilized in this study was sourced from a public domain. It comprises comprehensive student records structured in a tabular format with 4,424 rows and 37 columns that reflect individual student entries and various attributes respectively.

Each column in the dataset represents distinct student-related variables relevant to the research objectives. Notable features include Marital Status, Gender, Scholarship Holder, Debtor, Tuition Fees Up to Date, etc. Importantly, the dataset is fully complete and contains no missing (null) values which are important for subsequent analysis.

3.2. Data Preparation

The data preparation conducted the Exploratory Data Analysis (EDA) to gain initial insights into the dataset. Visual tools such as bar graphs, bar graphs with Kernel Density Estimation (KDE), box plots, and a correlation matrix were used to examine the datasets by analyzing its data distribution, detecting outliers, and understanding relationships between variables. These methods are foundational essential for guiding further analysis.

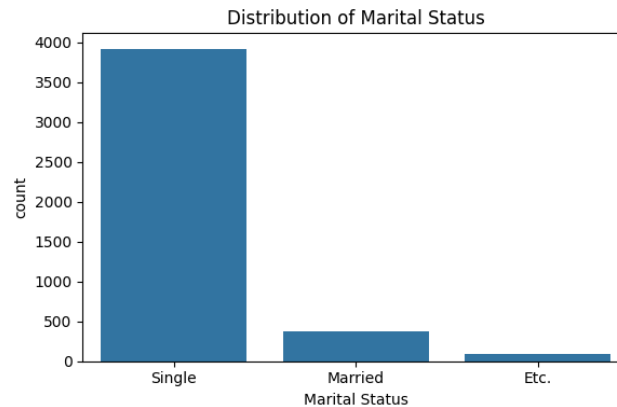
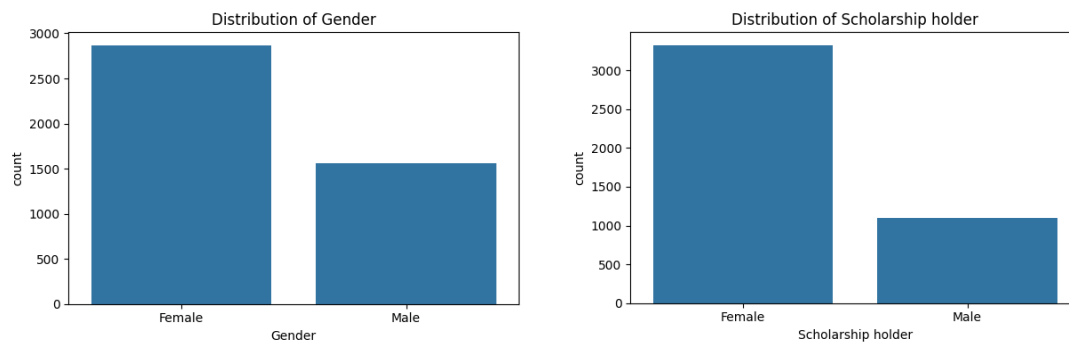


Figure 2. Distribution of Marital Status

Figure 2 shows the number of individuals categorized by their marital status. Most of the population is composed of single individuals, totaling 3,919, which represents approximately 88.6% of the entire group. This is followed by 379 individuals who are married, making up around 8.6% of the population. The smallest group consists of 91 widowed individuals, accounting for about 2.1%.



(A) Distribution of Gender

(B) Distribution of Scholarship Holder

Figure 3. Distribution of (A) Gender and (B) Scholar Holder

Figure 3 shows the population under study's gender and scholarship holder composition. According to the gender data, 2,868 people, or roughly 64.8% of the population, identify as female. By contrast, 1,556 people (approximately 35.2%) identify as male. The data for scholarship holder shows that 3,325 female students received scholarships, accounting for approximately 75.2% of the total scholarship holders. In contrast, 1,099 male students were recipients, representing about 24.8%.

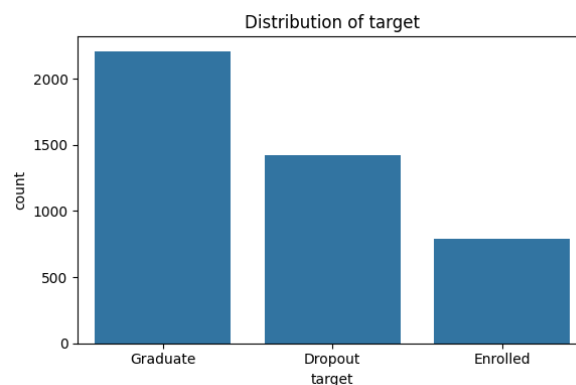


Figure 4. Distribution of Target (Dependent Variable)

Figure 4 shows the classification of individuals based on their current educational status. According to the data, 2,209 people, or roughly 49.9% of the total population, are graduates. 1,421 people, or roughly 32.1% of the total, are classified as dropouts after this. The smallest group comprises 794 enrolled individuals, or 18.0% of the total population.

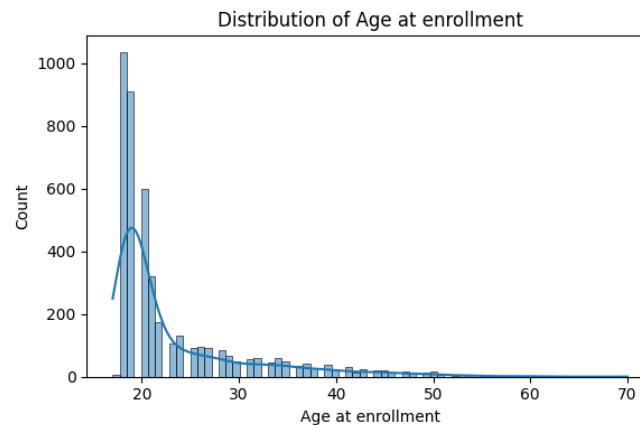


Figure 5. Distribution of Age

Figure 5 shows the frequency of individuals based on their age at the time of enrollment. The distribution is right-skewed and indicates that many enrollees were younger with a gradual decline in frequency as age increases. A large concentration of enrollments is observed between the ages of 17 to 22 and peaking around age 18, which is typical for students entering higher education immediately after secondary school. After age 22, the number of enrollees steadily decreases, though enrollment still occurs across a wide age range.

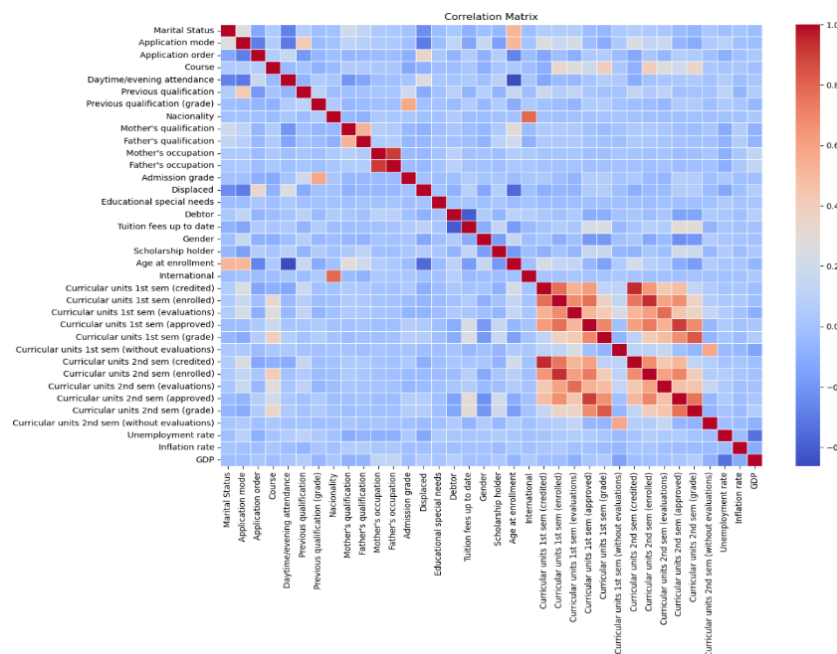


Figure 6. Correlation Matrix

Figure 6 shows the matrix of academic-related variables such as the number of units enrolled, approved, and corresponding grades in both the first and second semesters' exhibit strong positive correlations. This suggests that students who enroll in more units tend to pass more subjects and obtain higher academic performance and indicate consistency in student engagement and achievement. On the other hand, most background and demographic factors are nationality, gender, marital status, and

education. The employment of parents shows little to no association with academic performance. Similarly, there are weak correlations between academic performance indicators and financial status variables like tuition fee status, debtor classification, and scholarship grants.

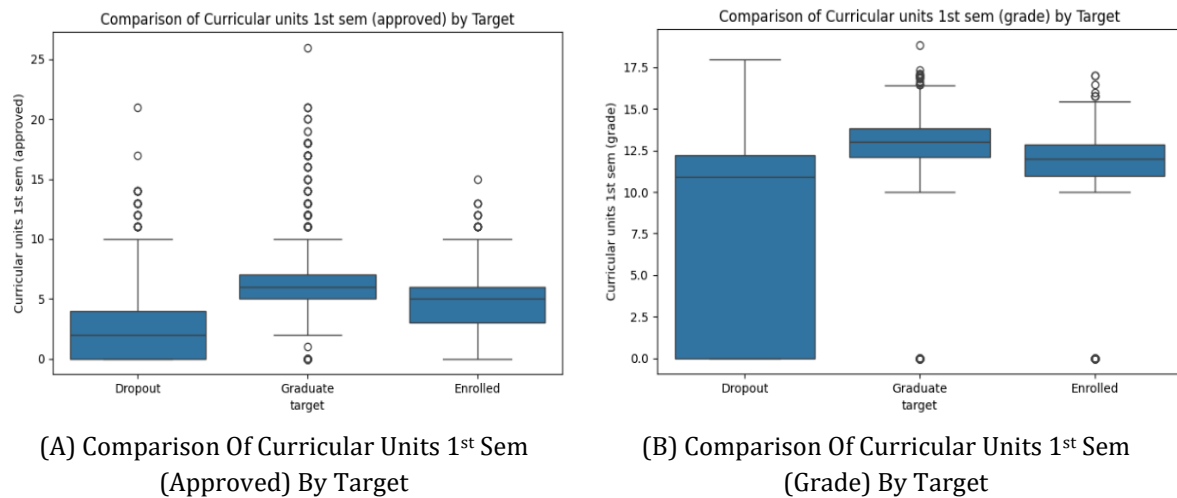


Figure 7. Comparison of Curricular Unit's 1st SEM for (A) Approved and (B) Grade by Target

Figure 7 shows a visual comparison between the number of approved and graded curricular units in the first semester. Graduates consistently exhibit the best academic performance, with median grades of 13–14 and the highest median number of approved units (approximately 6). They also show relatively narrow interquartile ranges (IQRs), indicating stable achievement. Closely behind are enrolled students who have somewhat lower medians and moderate unit and grade variability. In contrast, dropouts demonstrate the weakest performance, with a low median of approved units (around 2) and lower grades (median around 11), along with greater variability and numerous low outliers, including many who approved zero units.

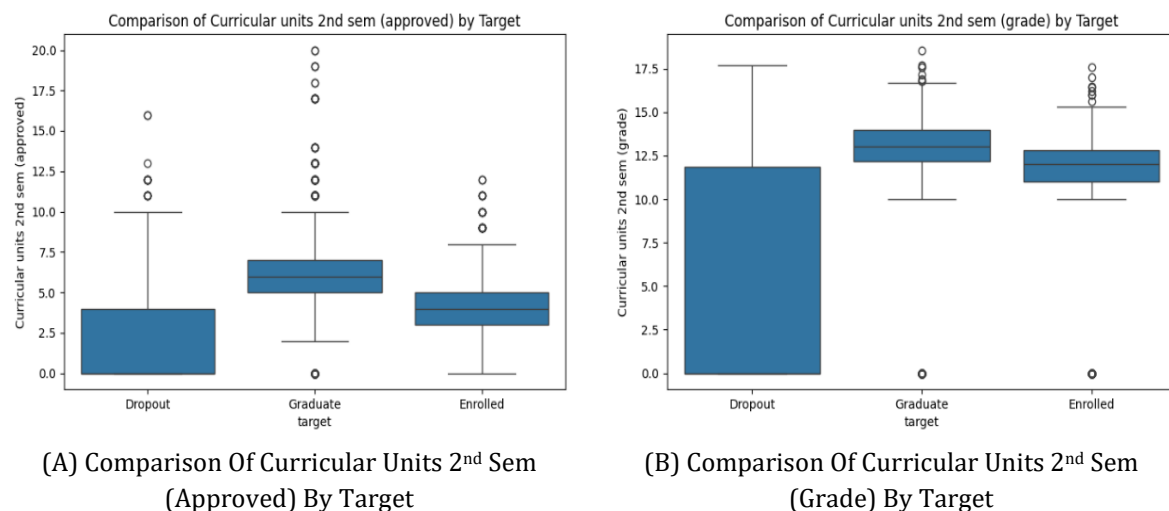


Figure 8. Comparison of Curricular Unit's 2nd SEM for (A) Approved and (B) Grade by Target

Figure 8 provides a graphic comparison of second semester approved and graded curriculum units. In terms of authorized units, the dropouts have a median close to zero that indicates a little academic progress while graduates have the highest median (about 6), followed by enrolled students (about 4). In contrast to dropouts who consistently perform poorly and have a narrow IQR, the graduates also show the widest IQR, indicating greater variability in achievement. Similarly, graduates and enrolled students maintain relatively compact IQRs and higher medians (around 13) when looking at grades, this indicates

consistent and improved academic performance. Dropouts by contrast, present a broader and lower distribution with many low grades including zeros and potentially reflecting failed or incomplete courses.

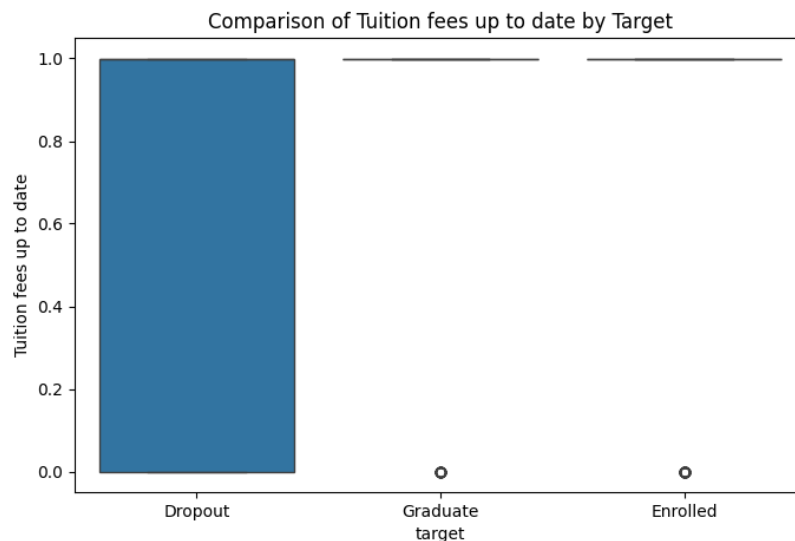


Figure 9. Comparison of Tuition Fees Up to Date by Target

Figure 9 shows a box plot that compares the status of tuition fee payments across three student.

Outcome Categories

Dropout, Graduate, and Enrolled. For all three groups, most students have their tuition fees up to date, as indicated by the box plots centered at 1. However, there are a few outliers at 0 in the Graduate and Enrolled categories, suggesting that some students in these groups had unpaid or delayed tuition. Interestingly, there are no such outliers in the Dropout group, indicating that all dropout students had their tuition fully paid.

3.3. Train the Model

In the model training phase, the categorical target variable was first converted by converting it into numbers using label encoding as most machine learning algorithms require numerical input because of their efficiency in managing numerical and categorical data, their interpretability, and their capacity to comprehend more intricate non-linear patterns on the tree-based algorithms were chosen. The tree-based algorithms were.

1. **Decision Tree Classifier:** Employs a hierarchical tree-like decision-making model in which each leaf node designates a class label and each internal node divides data according to a feature value. It is valued for its interpretability and clarity, which make the reasoning and predictions simple to comprehend [16].
2. **Random Forest Classifier:** Using bootstrapped samples and random feature selection, it creates an ensemble of decision trees. Then aggregates the votes from the trees to increase accuracy and decrease overfitting, and it provides metrics such as feature importance to aid in the interpretation of model decisions [17].
3. **Gradient Boosting Classifier:** Constructs a group of weak learners, usually decision trees in a step-by-step manner. Gradient descent on a given loss function (such as logistic loss) is essentially carried out at each step by training a new tree to predict the residual or error of the current ensemble. This iterative process continues, adding trees that correct previous errors, resulting in a powerful classifier that can deliver high accuracy, handle various data types, and provide feature importance metrics [18].
4. **Extra Trees Classifier:** Constructs multiple decision trees using the entire dataset (without bootstrapping) and determines split thresholds entirely at random. High levels of randomness in

feature selection and split point selection expedite training, lower variance, and frequently result in excellent generalization performance in classification tasks [19].

5. **Histogram-Based Gradient Boosting Classifier:** Uses histogram-based gradient boosting which discretizes features into bins to speed up split finding while maintaining accuracy to build an ensemble of decision trees. It continues to be very successful at classification tasks, offering quick training and robust results across a range of data kinds. [20].
6. **Cat Boost:** is an excellent gradient-boosting decision tree algorithm that uses ordered target statistics to handle categorical features natively and avoid target leakage. It delivers high accuracy, fast training, and strong performance on tabular data, especially with mixed numerical and categorical types. Studies show it often outperforms rivals like XGBoost and LightGBM in these settings [21].
7. **XGBoost:** is an optimized gradient boosting framework that iteratively builds an ensemble of decision trees using second-order gradients and regularization to enhance performance and prevent overfitting. It provides strong feature importance metrics for interpretability and excels in speed, scalability, and accuracy across structured datasets. [22].
8. **Light GBM:** is a quick and effective gradient boosting framework that uses exclusive feature bundling (EFB), gradient-based one-side sampling (GOSS), leaf-wise growth, and histogram-based tree learning to minimize computation and memory while preserving high accuracy. It provides interpretable feature importance metrics that scales to large datasets and performs exceptionally on tabular data. [23].

The dataset was divided into trains and test sets in an 80:20 ratio in order to evaluate the effectiveness of these algorithms. This indicates that 20% of the data was set aside for assessing the model's capacity for generalization and the remaining 80% was used for model training.

4. RESULTS AND DISCUSSION

4.1. Evaluate the Model

In this phase focuses on evaluating the performance of tree-based algorithms such as Decision Tree, Random Forest, Gradient Boosting, XGBoost, LightGBM, Extra Trees, CatBoost, and HistGradient Boosting are utilized to explore complex patterns and improve model accuracy using standard metrics. The metric uses the following:

1. **Classification Report:** offers a thorough synopsis of a classification model's performance. The F1-score, recall, and precision are usually included [24]. Additionally, weighted averages, which consider the number of instances per class to account for class imbalance and macro-averages which treat all classes equally regardless of their size, are included in the classification report [25].
2. **Confusion Matrix Report:** is a tabular representation of a classification model's performance that shows the actual versus predicted classifications. It is composed of four primary components: true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). These values help quantify the number of correct and incorrect predictions made by the model for each class [26].

Table 1. Comparison of Tree-Based Models Has Been Used

	Tree-Based Model	Accuracy	(In Percentage)	Rank
1	Decision Tree	0.6768	67.68%	8
2	Random Forest	0.7605	76.05%	7
3	Gradient Boosting	0.7616	76.16%	6
4	XGBoost	0.7695	76.95%	3
5	LightGBM	0.7729	77.29%	1
6	Extra Trees	0.7684	76.84%	4
7	CatBoost	0.7627	76.27%	5
8	HistGradient Boosting	0.7718	77.18%	2

Table 1 presents a comparative analysis of eight tree-based algorithms based on their accuracy. The models that performed the best were Light GBM (77.29%), Hist Gradient Boosting (77.18%), and XGBoost (76.95%). The Decision Tree (67.68 %) model had the lowest accuracy at the lower end.

4.2. Fine Tune the Model

This phase uses hyperparameter optimization methods like Grid Search CV, Randomized Search CV, and Bayes Search CV to improve the performance of tree-based models by methodically examining the parameter space to find the best configurations.

Table 2. After Applying for Randomizedsearchcv

	Tree-Based Model	Accuracy	(In Percentage)	Rank
1	Decision Tree	0.7437	74.37%	8
2	Random Forest	0.7783	77.83%	5
3	Gradient Boosting	0.7771	77.71%	6
4	XGBoost	0.7837	78.37%	1
5	LightGBM	0.7796	77.96%	3
6	Extra Trees	0.7764	77.64%	7
7	CatBoost	0.7814	78.14%	2
8	HistGradient Boosting	0.7796	77.96%	3

Table 2 shows the classification accuracy of various tree-based models after using hyperparameter tuning of Randomized Search CV. XG Boost (78.37%) was the best-performing model, closely followed by Cat Boost (78.14%). It's interesting to note that Light GBM and Hist Gradient Boosting both tied for third place with an accuracy of 77.96%. Because of its simpler architecture and lack of ensemble techniques, the Decision Tree (74.37 %) model continued to perform the worst even after optimization.

Table 3. After Applying for Gridsearchcv

	Tree-Based Model	Accuracy	(In Percentage)	Rank
1	Decision Tree	0.7437	74.37%	8
2	Random Forest	0.7755	77.55%	6
3	Gradient Boosting	0.778	77.80%	5
4	XGBoost	0.7837	78.37%	1
5	LightGBM	0.7807	78.07%	3
6	Extra Trees	0.7735	77.35%	7
7	CatBoost	0.7814	78.14%	2
8	HistGradient Boosting	0.7803	78.03%	4

Table 3 summarizes the classification accuracy of eight tree-based models after applying GridSearchCV for exhaustive hyperparameter tuning. The XGBoost (78.37%) model continued to lead the others, followed by CatBoost (78.14%) and LightGBM (78.07%). Interestingly, HistGradient Boosting (78.03%) and Gradient Boosting (77.8%) came respectively. Even after optimization, the Decision Tree (74.37%) model's accuracy indicated that it was limited in its ability to represent intricate patterns when compared to ensemble approaches.

Table 4. After Applying for Bayessearchcv

	Tree-Based Model	Accuracy	(In Percentage)	Rank
1	Decision Tree	0.7477	74.77%	8
2	Random Forest	0.7776	77.76%	6
3	Gradient Boosting	0.7798	77.98%	5
4	XGBoost	0.7814	78.14%	2

5	LightGBM	0.7812	78.12%	4
6	Extra Trees	0.7771	77.71%	7
7	CatBoost	0.7814	78.14%	2
8	HistGradient Boosting	0.7839	78.39%	1

Table 4 displays the classification accuracy of various tree-based models after hyperparameter optimization using BayesSearchCV. The HistGradient Boosting (78.39) outperformed both XGBoost and CatBoost which were tied for second place with 78.14%. While Random Forest (77.76%) and Gradient Boosting (77.98%) continued to perform competitively. Despite tuning, the Decision Tree (74.77%) ranked lowest as was to be expected.

4.3. Predictive Model

Displaying a multi-line graph that illustrates each model's accuracy under various tuning strategies. The effects of RandomizedSearchCV, GridSearchCV, and BayesSearchCV demonstrate the possible benefits of methodical hyperparameter optimization, while the original (untuned) tree-based model performances act as the baseline reference.

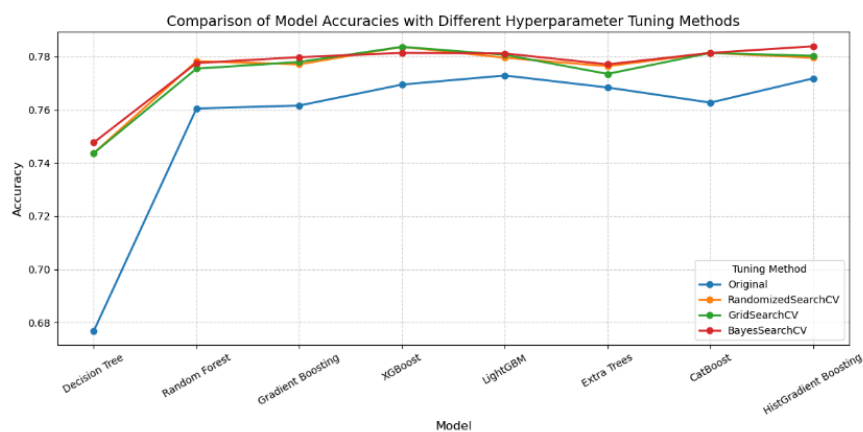


Figure 10. Accuracy Trends across Tuning Techniques for Tree Models

Figure 10 reveals that the XGBoost (78.14%) classifier continuously obtained the best accuracy among all algorithms, with the BayesSearchCV configuration showing a minor improvement over the initial baseline (78.37%). CatBoost and HistGradient Boosting also showed consistent performance, with HistGradient Boosting being the best-performing model with the highest score under BayesSearchCV (78.39%). When optimized with BayesSearchCV, the Decision Tree model, which had the lowest accuracy across all configurations, only slightly improved (from 74.37% to 74.77%), suggesting that it has a limited ability to gain from hyperparameter tuning in comparison to ensemble-based techniques.

Comparative findings indicate that the degree of improvement varies among the algorithms, hyperparameter tuning improves model performance. Out of all the tuning methods, BayesSearchCV typically produces marginally higher accuracy gains than RandomizedSearchCV and GridSearchCV, indicating its benefit in effectively exploring the hyperparameter space. Additionally, complex ensemble models such as XGBoost, LightGBM, and CatBoost demonstrate greater potential for improvement when subjected to tuning, in contrast to Decision Tree which show minimal change. This highlights the necessity of using suitable optimization techniques strategies to maximize the potential of machine learning algorithms in predictive modeling tasks.

5. CONCLUSION

The most reliable gains were produced by Bayes Search CV, especially for sophisticated ensemble models like XGBoost and HistGradient Boosting. These findings highlight how crucial it is to combine robust model architectures with optimization strategies to produce precise and trustworthy predictions. When

high performance is needed, it is advised that future predictive modeling initiatives use ensemble tree-based models after first implementing structured Exploratory Data Analysis. Moreover, incorporating hyperparameter tuning, especially Bayes Search CV should be considered standard practice to boost the model's effectiveness across multiple areas domains.

Acknowledgments

The researcher gratefully acknowledges Kaggle.com for providing access to publicly available datasets which were instrumental in the successful completion of this study.

Funding Information

This research did not receive any grant from funding agencies.

Author Contributions Statement

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Roman B. Villones	✓	✓	✓	✓	✓	✓			✓	✓	✓	✓	✓	

C : Conceptualization

M : Methodology

So : Software

Va : Validation

Fo : Formal analysis

I : Investigation

R : Resources

D : Data Curation

O : Writing - Original Draft

E : Writing - Review & Editing

Vi : Visualization

Su : Supervision

P : Project administration

Fu : Funding acquisition

Conflict of Interest Statement

The author declared no conflict of interest.

Informed Consent

The datasets came from a public domain platform that allows the use of data for research. Since there was no personally identifiable information in the data, informed consent was not required.

Ethical Approval

Ethical approval was considered unnecessary because the dataset was taken from a public domain platform intended for research purposes and contained no identifiable or sensitive data.

Data Availability

The study's dataset, "Student Dropout & Success Prediction Dataset" by Adil Shamim (Kaggle), is openly accessible at <https://www.kaggle.com/datasets/adilshamim8/predict-students-dropout-and-academic-success>. Although the dataset was not gathered or owned by the author, it was used for scholarly and research purposes under a public license.

REFERENCES

- [1] M. Nabil and A. Seyam, 'Predicting students' academic performance using machine learning techniques: a literature review', International Journal of Business Intelligence and Data Mining, vol. 20, no. 4, pp. 456-479, 2022. doi.org/10.1504/IJBIDM.2022.123214
- [2] M. A. Sagun, P. D. Soriano, J. R. Pedrasa, and D. Ong (2021). Modelling Student Dropout using AdaBoost and Survival Analysis. Philippine Engineering Journal, 42(2).
- [3] S. J. Parreño, 'School dropouts in the Philippines: causes, changes and statistics', Sapienza: International Journal of Interdisciplinary Studies, vol. 4, no. 1, pp. e23002-e23002, 2023. doi.org/10.51798/sijis.v4i1.552
- [4] E. A. Bilar and M. C. A. Montañez, 'Understanding student dropout risk: A qualitative study', HO CHI MINH CITY OPEN UNIVERSITY JOURNAL OF SCIENCE-SOCIAL SCIENCES, vol. 14, no. 2, pp. 18-34, 2024. doi.org/10.46223/HCMCOUJS.soci.en.14.2.2825.2024

- [5] M. Naseem, K. Chaudhary, and B. Sharma, 'Using ensemble decision tree model to predict student dropout in computing science', in 2019 IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), 2019, pp. 1-8. doi.org/10.1109/CSDE48274.2019.9162389
- [6] M. Nagy and R. Molontay, 'Interpretable dropout prediction: towards XAI-based personalized intervention', International Journal of Artificial Intelligence in Education, vol. 34, no. 2, pp. 274-300, 2024. doi.org/10.1007/s40593-023-00331-8
- [7] B. K. Baradwaj, and S. Pal (2022). STUDENT DROPOUT PREDICTION USING MACHINE LEARNING TECHNIQUES. International Journal of Advanced Research in Computer Science, 16(2), March-April 2025. doi.org/10.26483/ijarcs.v16i2.7209
- [8] Z. Zhao, and P. Ren (2025). Random Forest-Based Early Warning System for Student Dropout Using Behavioral Data. Bulletin of Education and Psychology.
- [9] W. Chow (2024, December). Improving early dropout detection in undergraduate students: Exploring key predictors through SHAP values. In Proceedings of the 35th Annual Conference of the Australasian Association for Engineering Education (AAEE 2024) (pp. 149-158). Christchurch, New Zealand: Engineers Australia.
- [10] A. Asselman, M. Khaldi, and S. Aammou, 'Enhancing the prediction of student performance based on the machine learning XGBoost algorithm', Interactive Learning Environments, vol. 31, no. 6, pp. 3360-3379, 2023. doi.org/10.1080/10494820.2021.1928235
- [11] G. Li, J. Cui, H. Fu, and Y. Sun, 'Light GBM and GA Based Algorithm for Predicting and Improving Students' Performance in Higher Education in the Context of Big Data', in 2024 7th International Conference on Education, Network and Information Technology (ICENIT), 2024, pp. 1-5. doi.org/10.1109/ICENIT61951.2024.00009
- [12] D. Kumar, A. Kothiyal, R. Kumar, C. Hemantha, and R. Maranan, 'Random Forest approach optimized by the Grid Search process for predicting the dropout students', in 2024 International Conference on Innovations and Challenges in Emerging Technologies (ICICET), 2024, pp. 1-6. doi.org/10.1109/ICICET59348.2024.10616372
- [13] N. Alamsyah, B. Budiman, T. P. Yoga, and R. Y. R. Alamsyah, 'XGBOOST HYPERPARAMETER OPTIMIZATION USING RANDOMIZEDSEARCHCV FOR ACCURATE FOREST FIRE DROUGHT CONDITION PREDICTION', Jurnal Pilar Nusa Mandiri, vol. 20, no. 2, pp. 103-110, 2024. doi.org/10.33480/pilar.v20i2.5569
- [14] S. Albahli, 'Efficient hyperparameter tuning for predicting student performance with Bayesian optimization', Multimed Tools Appl, vol. 83, pp. 52711-52735, 2024. doi.org/10.1007/s11042-023-17525-w
- [15] R. Makovetsky, N. Piche, and M. Marsh, 'Dragonfly as a platform for easy image-based deep learning applications', Microsc. Microanal., vol. 24, no. S1, pp. 532-533, Aug. 2018. doi.org/10.1017/S143192761800315X
- [16] Y. Shuo Tan, A. Agarwal, and B. Yu (2022). A cautionary tale on fitting decision trees to data from additive models: generalization lower bounds. arXiv e-prints, arXiv:2110. doi.org/10.48550/arXiv.2110.09626
- [17] M. Schonlau and R. Y. Zou, 'The random forest algorithm for statistical learning', Stata Journal, vol. 20, no. 1, pp. 3-29, 2020. doi.org/10.1177/1536867X20909688
- [18] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, 'A comparative analysis of gradient boosting algorithms', Artificial Intelligence Review, vol. 54, no. 8, pp. 1937-1967, 2021. doi.org/10.1007/s10462-020-09896-5
- [19] N. Hussein and S. R. M. Zeebaree, 'Performance evaluation of Extra Trees classifier by using CPU parallel and non-parallel processing', The Indonesian Journal of Computer Science, vol. 13, no. 2, 2024. doi.org/10.33022/ijcs.v13i2.3802
- [20] N.-D. Hoang and V.-D. Tran, 'Comparison of histogram-based gradient boosting classification machine, random forest, and deep convolutional neural network for pavement raveling severity classification', Automation in Construction, vol. 148, 2023. doi.org/10.1016/j.autcon.2023.104767

- [21] A. Odeh, Q. Al-Haija, A. Aref, and A. Abu Taleb, 'Comparative study of CatBoost, XGBoost, and LightGBM for enhanced URL phishing detection: A performance assessment', Journal of Internet Services and Information Security, vol. 13, no. 4, 2023. doi.org/10.58346/IJISIS.2023.I4.001
- [22] V. Zelli et al., 'Classification of tumor types using XGBoost machine learning model: a vector space transformation of genomic alterations', Journal of Translational Medicine, vol. 21, 2023. doi.org/10.1186/s12967-023-04720-4
- [23] J. Yan et al., 'LightGBM: Accelerated genomically designed crop breeding through ensemble learning', Genome Biology, vol. 22, 2021. doi.org/10.1186/s13059-021-02492-y
- [24] G. M. Foody, 'Challenges in the real-world use of classification accuracy metrics: From recall and precision to the Matthews correlation coefficient', PLOS ONE, vol. 18, no. 10, 2023. doi.org/10.1371/journal.pone.0291908
- [25] O. Rainio, J. Teuho, and R. Klén, 'Evaluation metrics and statistical tests for machine learning', Scientific Reports, vol. 14, no. 1, 2024. doi.org/10.1038/s41598-024-56706-x
- [26] Ž. Vujović, 'Classification model evaluation metrics', International Journal of Advanced Computer Science and Applications, vol. 12, no. 6, pp. 599-606, 2021. doi.org/10.14569/IJACSA.2021.0120670

How to Cite: Roman B. Villones. (2025). Optimizing tree-based algorithms for student dropout prediction: a comparative study. Journal of Artificial Intelligence, Machine Learning and Neural Network (JAIMLNN). 5(2), 35–47. <https://doi.org/10.55529/jaimlnn.52.35.47>

BIOGRAPHY OF AUTHOR



Roman B. Villones , is a faculty researcher and data science lecturer at the College of Informatics, Philippine Christian University. He holds a master's degree in Information Technology at Philippine Christian University and is currently pursuing his doctorate degree in Information Technology at La Consolacion University Philippines. He has authored several peer-reviewed publications on software engineering and now focuses on classification algorithms, model optimization, and educational data mining. Email: roman.villones@pcu.edu.ph, villones.roman@gmail.com