Vol.3 No.1 February 2026: 9-19
International Journal of Computer Science and Humanitarian AI (IJCSHAI)

eISSN: 3064-4372
DOI: 10.21512/ijcshaijournal.v3i1.14936

Comparative Analysis of Decision Tree, Random Forest, and XGBoost for Student Category Prediction

Rayson Calvianto Lim[1*], Harvianto[2]

[1,2]Computer Science Department, School of Computer Science,
Bina Nusantara University,
Jakarta, Indonesia 11480
rayson.lim@binus.ac.id; harvianto@binus.ac.id

***Abstract*—This study presents a comparative analysis of Decision Tree, Random Forest, and XGBoost for student category prediction using structured educational data from Independent School Batam. The increasing availability of educational data provides opportunities to apply machine learning techniques for student performance prediction and curriculum personalization. The models were trained and evaluated using 120 student records collected from Independent School Batam and were evaluated through an end-to-end machine learning workflow consisting of data preprocessing, feature engineering, model training, and evaluation. An anonymized dataset containing student demographic variables, academic indicators, learning behavior attributes, and teacher feedback scores was prepared using data cleaning, categorical transformation, normalization, and stratified train–test splitting to ensure consistency and reproducibility. Model performance was assessed using accuracy, precision, recall, and F1-score. Experimental results show that Random Forest achieved the highest accuracy of 0.8333, outperforming Decision Tree and XGBoost across the evaluation metrics. Per-class analysis showed that the Medium category was predicted most consistently, while the Low category remained the most challenging to identify due to limited representation, resulting in lower recall values. This study contributes a practical and reproducible baseline comparison of machine learning models for student category prediction using lightweight, open-source tools. The findings provide empirical guidance for educational institutions seeking accessible approaches to classify student categories and support structured educational data analysis in resource-limited environments, as well as proposing a lightweight machine learning pipeline suitable for schools with limited technical infrastructure.**

Keywords*— *Machine Learning; Student category prediction; Random Forest; XGBoost; Descision Tree

I. Introduction

The integration of artificial intelligence (AI) and machine learning (ML) into educational systems has created new opportunities for improving the quality and personalization of learning [1]. In particular, the concept of personalized curriculum design, where teaching content and pacing are adapted to individual students' performance and learning styles has emerged as a central focus in modern educational research [2]. As global educational institutions move toward data-driven instruction under the framework of Education 4.0, the ability to use learning analytics effectively becomes a key differentiator in fostering student engagement and academic success [3].

In Indonesia, many schools have begun adopting digital learning environments, but the majority still lack the analytical capability to interpret and utilize educational data effectively [4]. Independent School Batam, located in Komplek Rosedale Blk. E No. 123–124, Teluk Tering, Batam Kota, Kepulauan Riau, is an example of a mid-sized institution that integrates both traditional and digital approaches through a blended learning model. Although the school employs a learning management system (LMS) to record student grades, attendance, and activity logs, the process of curriculum adjustment remains largely manual. Teachers must rely on intuition and experience to identify students with different performance levels, which limits the consistency and timeliness of academic evaluation [5]. This situation highlights a significant gap between data availability and its analytical use. While a large volume of student performance data exists within the school database, it is not systematically processed to generate structured insights that could support objective performance analysis [6]. Consequently, student category prediction at Independent School Batam remains dependent on manual interpretation rather than data-driven methods. This gap forms the primary motivation for this research: to explore whether foundational machine learning algorithms can be implemented in a practical and cost-effective manner to support systematic student category prediction.

The goal of this study is to conduct a comparative analysis of baseline machine learning models for classifying student performance levels. Specifically, this research compares the performance of three widely used baseline ML models: (1)Decision Tree, (2)Random Forest, and (3)XGBoost. These models are chosen due to their interpretability, relatively low computational requirements, and proven effectiveness in

Received: Dec. 12, 2025; received in revised form: Mar. 16, 2026; accepted: Mar. 18, 2026; available online: Mar. 29, 2026.

Corresponding: rayson.lim@binus.ac.id

educational data mining applications [7]. The findings are expected to provide a reference for schools with limited resources to adopt machine learning techniques for student category prediction without relying on complex or proprietary systems. To ensure accessibility and reproducibility, this research adopts a practical workflow using open-source tools. The implementation relies on Python (version 3.12) as the main programming environment, with Pandas and Scikit-learn libraries for data preprocessing and model development [8]. Student data are stored and managed through PostgreSQL (version 18), while Prefect (version 3) is employed as a workflow orchestration tool to automate data extraction, transformation, and loading processes [9]. Data exploration and visualization are conducted using JupyterLab, allowing clear and interactive analysis of learning trends [10]. The dataset, drawn from the database records of Independent School Batam, includes anonymized student identifiers, subject scores, participation frequencies, and assessment results. After cleaning and normalization, these data serve as input for the ML models to predict learning performance categories [11]. By applying these three baseline models, the research aims to identify which algorithm yields the most balanced trade-off between predictive accuracy and implementation simplicity. Evaluation metrics such as accuracy, precision, recall, and F1-score will be used to assess model performance. The overall framework emphasizes practicality over complexity, ensuring that even institutions without specialized data science expertise can replicate the process and benefit from its insights. This approach aligns with the growing need for educational innovations that are not only technologically advanced but also pedagogically relevant and operationally sustainable [12].

In summary, this research presents a comparative machine learning-based approach to student category prediction using data from Independent School Batam. It demonstrates how small to medium-sized schools can adopt data-driven strategies to analyze student performance without significant technical or financial investment. The remainder of this paper is structured as follows: the next section reviews relevant literature and theoretical foundations of educational data mining and machine learning models; the third section describes the methodology and implementation process; the fourth section presents and discusses the experimental results, and the final section provides conclusions and recommendations for future work.

II. Literature Review

Personalized learning has emerged as a central theme in modern education, emphasizing that teaching and learning should adapt to the unique needs, pace, and preferences of each student. The theoretical foundation of personalized learning draws from constructivist and learner-centered theories, which argue that knowledge is actively constructed through experience and reflection rather than passively received [13]. Effective learning occurs when instruction is tailored to the learner's cognitive readiness, prior knowledge, and social context [14]. In contemporary classrooms, these ideas have evolved into data-driven personalization strategies powered by digital technologies and analytics.

The integration of data analytics into education has given rise to the fields of Educational Data Mining (EDM) and Learning Analytics (LA), which aim to uncover patterns in student data to improve teaching and learning [15]. Learning analytics systems analyze student behaviors, assessments, and digital interactions to generate insights about performance, engagement, and learning trajectories [16]. Studies have shown that these approaches can support adaptive feedback, identify at-risk learners, and optimize curriculum sequencing [17]. However, these benefits depend heavily on the quality of collected data, the interpretability of the models used, and the alignment between analytics outputs and actual pedagogical decisions [18].

Machine Learning (ML) has become a crucial technique within educational data mining for modeling and predicting student performance, engagement, and behavior [19]. Common ML applications in education include predicting academic outcomes, recommending personalized study materials, and identifying learning difficulties early [20]. These systems rely on supervised learning models- algorithms that learn patterns from labeled data to predict future outcomes. Among the various ML approaches, tree-based models such as Decision Trees, Random Forests, and Gradient Boosting Machines (e.g., XGBoost) have been widely used due to their balance of accuracy, interpretability, and computational efficiency [21]. Studies have demonstrated that tree-based algorithms can effectively handle categorical and numerical data, identify key performance indicators, and visualize decision logic for educators [22].

Decision Tree algorithms work by recursively splitting datasets into homogeneous subgroups based on predictor variables, producing models that are easy to interpret and visualize [23]. Random Forest improves upon this by aggregating multiple Decision Trees to reduce overfitting and improve predictive [24] . XGBoost extends these principles through gradient boosting, allowing the model to iteratively correct errors from previous trees [25]. Comparative analyses indicate that Random Forest and XGBoost often outperform simpler models on structured tabular data, making them ideal baselines for category prediction tasks where interpretability remains essential [26].

A key insight from the literature is that the success of ML in educational contexts depends more on data engineering and pipeline design than on model complexity alone [27]. Constructing robust data pipelines is essential for maintaining data integrity and ensuring that models generalize effectively to future student cohorts [28]. Lightweight, modular pipelines using tools like Pandas, PostgreSQL, and Prefect can facilitate this process even in small institutional settings.

Equally important are the ethical and privacy considerations surrounding ML in education. Transparency, fairness, and data protection must be prioritized when analyzing student data [29]. Bias in datasets or algorithms can unintentionally disadvantage certain groups, while poor transparency may reduce trust among stakeholders.

Despite extensive global research on machine learning applications in education, significant gaps remain in its implementation in small and mid-sized schools with limited technical resources. Most existing studies focus on large-scale university systems or online learning platforms that have access to extensive datasets and advanced analytical infrastructure. Consequently, many smaller institutions lack access to practical machine learning solutions that can be implemented within their resource constraints. In addition, previous research often emphasizes improving prediction accuracy using large educational datasets, while relatively few studies explore lightweight machine learning pipelines that can be implemented in resource-constrained environments. Furthermore, predictive models are frequently evaluated only in terms of accuracy without clearly connecting the results to curriculum design or instructional decision-making. Therefore, this study aims to evaluate several machine learning algorithms for student performance prediction and investigate how predictive insights can support personalized curriculum planning in a school environment.

This study addresses that need by implementing and comparing three baseline ML algorithms, which consists of Decision Tree, Random Forest, and XGBoost. Within a lightweight, reproducible data pipeline tailored to Independent School Batam, grounded in existing research, this approach prioritizes interpretability, simplicity, and ethical responsibility while exploring how small-scale schools can leverage accessible technologies such as JupyterLab, Scikit-learn, and PostgreSQL for future machine learning research in the educational sector. The literature thus provides both the theoretical foundation and methodological guidance for designing an adaptable ML framework that bridges the gap between pedagogical theory and practical implementation in real-world educational contexts.

III. Proposed Methods

There are six essential stages in developing a machine learning-based personalized curriculum system: data preprocessing, feature engineering, model selection, model training, and evaluation. Each stage contributes to building a lightweight and reproducible ML pipeline that aligns with the technical and ethical standards of educational data analysis.

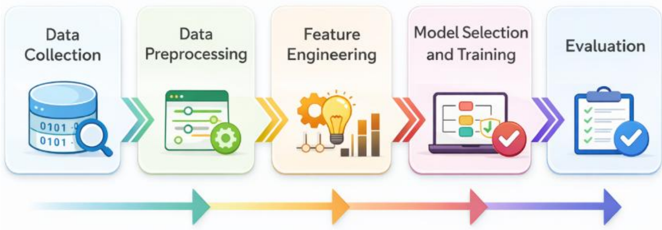

Figure 1. Research methodology flowchart

A. Data Collection

Data collection involves gathering academic and behavioral information from students [30]. The dataset was prepared and anonymized to ensure data privacy while maintaining the representativeness of student learning patterns [31].

Table 1. Sample Dataset Structure

Category	**Sample Attributes**
Student Demographics	Age, gender, grade level
Academic Performance	Exam scores, assignment completion, subject grades
Learning Behaviors	Attendance rate, class participation
Curriculum Attributes	Subject difficulty level, teacher feedback

Table 2. Sample Dataset Records

Student ID	**Age**	**Gender**	**Grade Level**	**Exam Score**	**Assignment Completion (%)**	**Attendance Rate (%)**	**Participation Level**	**Subject Difficulty**	**Teacher Feedback**
ISB001	14	Male	Grade 8	78	85	92	Medium	Moderate	Satisfactory
ISB002	15	Female	Grade 9	88	95	96	High	High	Very Good
ISB003	14	Female	Grade 8	70	80	88	Low	Low	Needs Improvement

Table 2 illustrates the structure of the dataset used in this study by presenting three representative student records. Each row corresponds to an anonymized student entry, while each column represents a demographic, academic, behavioral, or curriculum-related variable. The dataset combines performance indicators, learning behaviors, and curriculum attributes to provide structured input for training and evaluating machine learning models for all three models used. Teacher feedback was initially collected in qualitative form and later standardized for analysis.

B. Data Preprocessing

Data preprocessing converts raw input into a structured, consistent form suitable for ML algorithms. This includes [32]:

1. Cleaning

   During the initial data extraction phase, the dataset contained a larger number of records than those ultimately used for analysis [33]. Specifically, the raw extraction process yielded 194 records, as the school information system stores data at an event and activity level rather than a consolidated student level. As a result, multiple records could correspond to the same student due to separate entries for examinations, assignments, attendance logs, or administrative updates.

   Data cleaning aimed to remove inconsistencies and improve the reliability of the dataset [34]. Missing values in numerical attributes such as exam scores, assignment completion, attendance rate, and participation level, along with records with multiple missing fields were removed. Duplicate student records were identified using anonymized identifiers and eliminated. Additionally, value range checks were applied to ensure that numerical data fell within valid academic limits, reducing noise caused by data entry errors.

   During data cleaning, these raw records were reviewed and consolidated into a single representative record per student. Duplicate entries, incomplete records, and records lacking sufficient information across key attribute groups were removed. After this filtering and consolidation process, the dataset was reduced to 120 complete and reliable student records, which were then used for model training and evaluation.

Table 3. Dataset before cleaning

Student ID	Age	Gender	Grade Level	Exam Score	Assignment Completion (%)	Attendance Rate (%)	Participation Level	Subject Difficulty	Teacher Feedback
ISB001	14	Male	Grade 8	78	85	92	Medium	Moderate	Good
ISB002	15	Female	Grade 9	85	95	96	High	High	Very Good
ISB003	14	Female	Grade 8	80	80	88	Low	Low	Needs Improvement
ISB004	14	Male	Grade 8	78	85	92	Medium	Moderate	Good

In the raw dataset, some records contained logical errors such as attendance values exceeding 100%, missing assignment completion values, or duplicate entries for the same student. When these records were identified, it was then cleaned. Qualitative teacher feedback was standardized and mapped to numerical scores to support machine learning analysis.

Teacher feedback was originally recorded in qualitative form using descriptive labels. During preprocessing, these labels were standardized and converted into numerical scores based on a predefined mapping scheme. This transformation ensured consistency and allowed teacher feedback to be treated as a continuous variable during model training.

Table 4. Dataset after cleaning

Student ID	Age	Gender	Grade Level	Exam Score	Assignment Completion (%)	Attendance Rate (%)	Participation Level	Subject Difficulty	Teacher Feedback
ISB001	14	Male	Grade 8	78	85	92	Medium	Moderate	Good
ISB002	15	Female	Grade 9	85	95	96	High	High	Very Good

The values shown after data cleaning represent retained and standardized records following the removal of incomplete and duplicate entries, rather than direct modification of the same raw rows.

2. Encoding

Categorical attributes, including gender, grade level, subject difficulty, and teacher feedback, were transformed into numerical form using one-hot encoding. This method was selected to prevent the introduction of unintended ordinal relationships between categories while ensuring compatibility with all machine learning models. Encoding was performed after data cleaning to ensure that only valid and consistent category values were included in the dataset.

Table 5. Sample Categorical Encoding

Gender	Participation Level	Subject Difficulty
Male	Medium	Moderate
Female	High	High
Male	Low	Low

Table 6. After Categorical Encoding

Gender_Male	Gender_Female	Part_Low	Part_Medium	Part_High	Diff_Low	Diff_Moderate	Diff_High
1	0	0	1	0	0	1	0
0	1	0	0	1	0	0	1
1	0	1	0	0	1	0	0

3. Normalization

Feature normalization was applied to numerical attributes such as age, exam scores, assignment completion, attendance rate, and participation level to ensure comparable value ranges across features. Min–Max normalization was used to scale values between 0 and 1, preventing attributes with larger numeric ranges from dominating the learning process. Although tree-based models are less sensitive to scaling, normalization ensured consistency across all evaluated models, including XGBoost.

Table 7. Before Normalization

Exam Score	Assignment Completion	Attendance Rate	Teacher Feedback Score
78	85	92	3.5
88	95	96	4.2
70	80	88	2.8

Table 8. After Normalization

Exam Score	Assignment Completion	Attendance Rate	Teacher Feedback Score
0.53	0.67	0.67	0.63
0.80	1.00	1.00	0.85
0.00	0.33	0.33	0.00

In this study, Min–Max normalization was applied to numerical attributes. This method rescales values based on the minimum and maximum values observed in the dataset, not on a fixed scale such as 0–100 [35].

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

Where:
X= original value
$X_{\min}$= minimum value of the feature in the dataset
$X_{\max}$= maximum value of the feature in the dataset

4. Splitting

After completing data cleaning, encoding, and normalization, the final dataset consisted of 120 student records with the target variable performance_category.

Table 9. After Preprocessing

Student ID	Exam Score	Attendance Rate	Engagement Index	Performance Category
ISB001	0.53	0.67	0.62	High
ISB002	0.80	1.00	0.88	Medium
ISB003	0.45	0.55	0.50	Low
…	…	…	…	…

The dataset was divided using a stratified train–test split with the following ratio:

1. 80% training data (96 records)
2. 20% testing data (24 records)

Stratified splitting was applied to ensure that the proportions of High, Medium, and Low performance categories were preserved in both the training and testing datasets [36]. This approach prevents class imbalance issues, particularly for minority classes, and ensures a fair and representative evaluation of model performance.

Table 10. Reserved Class Distribution

Performance Category	Training Set	Testing Set
High	Proportional	Proportional
Medium	Proportional	Proportional
Low	Proportional	Proportional

Stratified splitting was used to maintain balanced class distribution between training and testing data, ensuring fair model evaluation.

C. Feature Engineering

Feature engineering transforms raw data into meaningful attributes that influence learning outcomes [37]. Feature engineering plays a crucial role in machine learning by transforming raw data into informative attributes that better represent underlying patterns related to student performance. Rather than relying solely on individual raw variables, feature engineering was applied in this study to summarize key academic and behavioral characteristics into compact and meaningful indicators. This approach improves model interpretability, reduces noise, and enhances the ability of machine learning models to distinguish between different student performance categories.

1. *Average score per subject area*

   (ExamScore+AssignmentCompletion) / 2
2. *Engagement index*

   (combining attendance + activity)
3. *Consistency ratio*

   (Var(ExamScore , AssignmentCompletion))

The average score per subject area represents a consolidated measure of academic performance. It was calculated by combining exam scores and assignment completion values using a simple arithmetic mean, as shown above. This aggregation provides a balanced representation of both summative assessment (exams) and formative assessment (assignments), reducing reliance on a single performance indicator. By integrating these two components, the resulting feature captures a student's overall academic standing within a subject more effectively than individual raw scores.

The engagement index was designed to reflect student involvement in the learning process by combining behavioral indicators. Attendance rate and participation level were normalized and aggregated to form a single engagement score. Attendance reflects physical or virtual presence in learning activities, while participation level captures active involvement during instruction. Combining these variables produces a more holistic representation of student engagement than either feature alone, allowing the models to better identify patterns associated with consistent learning behavior.

The consistency ratio was introduced to capture the stability of a student's academic performance over time. This feature was computed by measuring the variance of available performance scores for each student. Lower variance indicates more consistent performance, while higher variance reflects fluctuations that may signal learning difficulties or irregular study habits. Although the dataset did not include long-term longitudinal records, this feature provided a limited but meaningful measure of performance consistency within the available data.

Overall, the feature engineering process enhanced the representational quality of the dataset while maintaining simplicity and interpretability. The engineered features reduced dimensionality, minimizing redundancy among raw variables, and supported more effective learning by Decision Tree, Random Forest, and XGBoost models. In addition, these features offer intuitive insights for educators, as they align closely with common educational concepts such as achievement level, engagement, and learning stability.

D. Model Selection and Training

Three baseline machine learning models was selected with the following reasons provided.

a. Decision Tree (DT) - simple, interpretable, suitable for small datasets [38].
b. Random Forest (RF) - ensemble method improving accuracy by reducing overfitting [38].

c. XGBoost (XGB) - optimized gradient boosting model with high performance in structured data [39].

Model training will use Scikit-learn (for DT and RF) and XGBoost library for XGB implementation. All models are trained using 10-fold cross-validation to ensure stability and fairness across model comparisons [40]. Hyperparameters such as learning rate (for XGBoost), tree depth, and estimator count will be optimized manually for simplicity and reproducibility [41].

Table 11. Hyperparameter Configuration of Machine Learning Models

Model	Hyperparameters
Decision Tree	random_state = 42
Random Forest	n_estimators = 100, random_state = 42
XGBoost	n_estimators = 200, learning_rate = 0.1, max_depth = 5, subsample = 0.9, colsample_bytree = 0.9, random_state = 42, eval_metric = mlogloss

The hyperparameters for each model were defined based on standard configurations and adjusted where necessary to ensure stable training performance.

The input variables for all models consisted of demographic, academic, behavioral, and curriculum-related features, including age, grade level, exam scores, attendance rate, participation level, subject difficulty, and teacher feedback. The target variable was student performance category, classified into three classes: High, Medium, and Low. Decision Tree, Random Forest, and XGBoost models were trained using default or lightly adjusted parameters provided by their respective libraries, with fixed random seeds to ensure reproducibility. This approach aligns with the study's objective of evaluating baseline model performance within a lightweight and practical implementation framework.

E. Evaluation

We chose to evaluate our model using four performance metrics: Accuracy, Precision, Recall and F1 Score, with formulas as follow [42]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

Where:
TP = True Positive
TN = True Negative
FP = False Positive
FN = False Negative

Accuracy measures overall correctness, precision identifies prediction reliability, recall reflects sensitivity to actual positives, and F1-score combines both into a single balanced metric. The model with the best balance across these metrics will be proposed as the optimal baseline for future researches in the educational sector.

IV. Experimental Results

This chapter presents the experimental results obtained from evaluating three baseline machine learning models; Decision Tree, Random Forest, and XGBoost for student category prediction at Independent School Batam. Each model was trained and tested using the same cleaned and preprocessed dataset to ensure fairness and comparability across evaluations. The results are reported using the evaluation metrics defined in the methodology section: accuracy, precision, recall, and F1-score. In addition, per-class classification reports are provided to show model behavior for each outcome category. The dataset was obtained from school academic records and anonymized before analysis to protect student privacy.

The final dataset used in the experiment consisted of 120 anonymized student records containing demographic attributes, academic indicators, learning behavior variables, and teacher-based feedback measures. The dataset used in this study was collected from Independent School Batam and consists of 120 student records. Each record contains several attributes representing student demographic characteristics, academic performance, and learning behavior. The dataset includes variables such as student ID, age, gender, grade level, exam score, assignment completion percentage, attendance rate, participation level, subject difficulty, and teacher feedback score. These features were selected to capture multiple aspects of the learning process, including academic achievement, engagement, and instructional factors. The dataset was used as input for training and evaluating machine learning models to classify student performance levels and support personalized curriculum planning.

Before model training, the dataset underwent preprocessing steps including data cleaning, handling categorical variables, and normalization where necessary to ensure compatibility with the machine learning algorithms.

After preprocessing, the target variable consisted of three outcome categories: High, Medium, and Low. A stratified train–test split was applied to preserve class proportions, and fixed random seeds were used to support reproducibility. Although the results presented in this section are based on the held-out test set, the training process incorporated cross-validation to promote stability across training partitions.

Table 12. Decision Tree Classification Report

Class	Precision	Recall	F1-score	Support
0	0.71	1.00	0.83	5
1	0.80	0.67	0.73	6
2	0.83	0.77	0.80	13
Accuracy	-	-	**0.79**	24
Macro Avg	0.78	0.81	0.79	24
Weighted Avg	0.80	0.79	0.79	24

Table 13. Random Forest Classification Report

Class	Precision	Recall	F1-score	Support
0	1.00	0.80	0.89	5
1	1.00	0.50	0.67	6
2	0.76	1.00	0.87	13
Accuracy	-	-	**0.83**	24
Macro Avg	0.92	0.77	0.81	24
Weighted Avg	0.87	0.83	0.82	24

Table 14. XGBoost Classification Report

Class	Precision	Recall	F1-score	Support
0	0.83	1.00	0.91	5
1	0.75	0.50	0.60	6
2	0.79	0.85	0.81	13
Accuracy	—	—	**0.79**	24
Macro Avg	0.79	0.78	0.77	24
Weighted Avg	0.79	0.79	0.78	24

The classification results also show variations in prediction performance across different student performance categories. For example, the Random Forest model achieved a perfect precision score of 1.00 for classes 0 and 1, indicating that predictions for these categories were highly reliable when they occurred. However, the recall value for class 1 was lower, suggesting that some students in this category were misclassified. This pattern may be influenced by the relatively small number of samples available for certain classes, which can make it more difficult for the model to learn stable patterns.

The results focus on the performance metrics outlined in Chapter 3, namely accuracy, precision, recall, and F1-score. Following the presentation of results, a discussion subsection interprets the findings in relation to the research objectives and existing literature. The evaluation results indicate that the Random Forest model achieved the best overall performance among the evaluated algorithms. The model obtained an accuracy of 0.83 and an F1-score of 0.82, outperforming both the Decision Tree and XGBoost models. This result suggests that ensemble learning techniques are more effective in handling structured educational datasets where multiple variables influence student performance outcomes.

The final dataset used in this experiment consisted of 120 anonymized student records containing demographic, academic, behavioral, and curriculum-related attributes, following the structure outlined in Chapter 3. After preprocessing, the target variable included three performance categories: High, Medium, and Low. The distribution of the categories reflects a typical school setting, with most students falling into the medium performance range. Data cleaning procedures ensured that no missing, duplicate, or inconsistent values remained. All categorical features were encoded using one-hot encoding, while numerical features were standardized to maintain scale consistency across models.

To ensure data reliability, a stratified train-test split was used to preserve class proportions. Reproducibility was achieved by fixing random seeds during training. Although the results displayed below derive from the test set, the training phase involved 10-fold cross-validation to promote stability across repeated data partitions, aligning with the evaluation design described in the methodology section. Collectively, these procedures ensured that the dataset used for experimentation was valid, consistent, and suitable for supervised learning tasks.

Table 12 to 14 presents a summary of the evaluation metrics for all three machine learning models. Based on the accuracy, precision, recall, and F1-score results, the Random Forest model demonstrated superior performance compared to Decision Tree and XGBoost.

Table 15. Overall Performance of the Three Machine Learning Models

Model	Accuracy	Precision	Recall	F1-score
Decision Tree	0.79	0.80	0.79	0.79
Random Forest	0.83	0.87	0.83	0.82
XGBoost	0.79	0.79	0.79	0.78

These results show that the Random Forest achieved the highest performance across all evaluation metrics, followed by the Decision Tree and XGBoost models. Random Forest achieved the best performance because it produced the highest accuracy (0.83) and the best balance of precision, recall, and F1-score compared to Decision Tree and XGBoost.

The superior performance of the Random Forest model can be attributed to its ensemble learning mechanism, which combines multiple decision trees to produce more stable predictions. Unlike a single decision tree that may overfit the training data, Random Forest reduces variance by aggregating predictions from many trees. This capability allows the model to capture more complex relationships between variables such as attendance rate, exam score, and assignment completion, which are important indicators of student learning performance.

Table 16 displays the detailed classification results for each model, organized by performance category. The table includes precision, recall, and F1-score for each class label, providing further insight into how well each model distinguished High-, Medium-, and Low-performing students.

Table 16. Per-Class Evaluation Metrics for Each Model

Model	Class	Precision	Recall	F1-score	Support
Decision Tree	High	0.71	1.00	0.83	5
	Low	0.80	0.67	0.73	6
	Medium	0.83	0.77	0.80	13
Random Forest	High	1.00	0.80	0.89	5
	Low	1.00	0.50	0.67	6
	Medium	0.76	1.00	0.87	13
XGBoost	High	0.83	1.00	0.91	5
	Low	0.75	0.50	0.60	6
	Medium	0.79	0.85	0.81	13

It can also be observed that the dataset contains a larger number of samples in class 2 compared to the other classes.

As a result, the models tend to perform better in predicting this category because more training examples are available for learning its patterns. This imbalance in class distribution may influence the predictive performance of the models and should be considered when interpreting the results.

In all three models, the Medium class received the highest support because it represented the majority of student records. The High and Low classes, being smaller subsets, exhibited greater variation in recall values across models.

To highlight patterns and trends across the outcome categories, several figures were generated from the evaluation metrics. The result illustrates the comparative accuracy of the three models, clearly showing Random Forest as the strongest performer. The result also presents the precision, recall, and F1-score of each model, allowing clearer visualization of how consistently each algorithm predicted across all outcome categories. Confusion matrices were also generated to provide deeper insight into model misclassifications and to show how each model performed when predicting specific outcome levels. These figures collectively support the numerical findings and allow clearer observation of model behavior across the different student outcome categories.

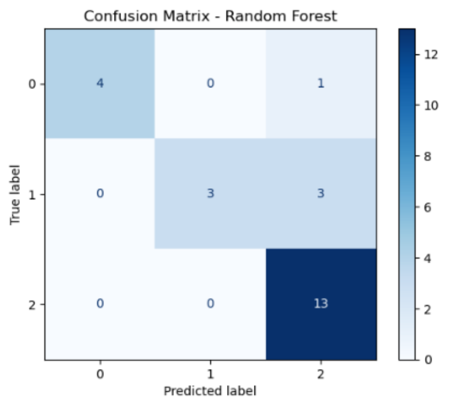

Figure 2. Random Forest Confusion Matrix

Figure 2. presents the confusion matrix for the Random Forest model, illustrating how the model classified student performance categories. The results show that the model correctly classified 4 out of 5 instances in class 0, with one instance misclassified as class 2. For class 1, the model correctly predicted 3 instances, while the remaining 3 were misclassified as class 2. The model demonstrated the strongest performance for class 2, correctly classifying all 13 instances without misclassification. These results indicate that the model is particularly effective in identifying students in the highest performance category, while some confusion occurs between the middle and higher performance categories. This may be attributed to similarities in feature patterns among these groups, such as comparable exam scores, attendance rates, or assignment completion levels.

The experimental results demonstrate that Random Forest produced the most reliable predictions for student category prediction. This model consistently outperformed both Decision Tree and XGBoost in overall accuracy and F1-score, suggesting that its ensemble-based structure enabled stronger generalization across the varied demographic, academic, and learning behavior features in the dataset. In contrast, Decision Tree and XGBoost achieved similar levels of predictive effectiveness but did not match the stability and overall performance produced by Random Forest. These results are consistent with prior research indicating that ensemble methods frequently outperform single-tree baselines in educational data mining tasks, as they reduce overfitting and capture complex relationships among features more effectively.

Although the results demonstrate promising predictive performance, it should be noted that the dataset used in this study contains only 120 student records. While the models were able to identify meaningful patterns within this dataset, the relatively small sample size may limit the generalizability of the results to larger educational contexts. Future research could incorporate larger datasets and additional learning variables to further validate the effectiveness of the proposed approach.

Further analysis reveals that the Medium outcome category was predicted with the highest consistency, which is expected due to its larger representation in the dataset. Conversely, the Low outcome category presented the greatest challenge for all models, as reflected in lower recall values, indicating that some low-outcome students were misclassified into other categories. This pattern is commonly observed in classification problems where minority classes have fewer representative samples for training, resulting in higher misclassification rates. The difficulty in identifying low-outcome students with high recall is particularly important for schools, as these students are typically those who require earlier academic support and closer monitoring. This finding reinforces the need for larger, more diverse datasets and potentially the inclusion of longitudinal indicators to improve sensitivity in identifying students in lower outcome categories.

When comparing the three models, the Decision Tree and XGBoost models achieved similar accuracy values of 0.7917. However, both models were slightly outperformed by Random Forest in terms of overall accuracy and F1-score. This finding suggests that ensemble-based methods may provide a more robust baseline for student performance prediction tasks, particularly when working with structured educational datasets containing multiple feature types.

Connecting these findings to the broader motivation discussed in the Introduction, this study confirms that lightweight machine learning pipelines can support data-driven educational analysis in resource-limited schools. The strong performance of Random Forest demonstrates that open-source, non-complex models can yield meaningful categorization insights and help educators identify learning patterns and potential academic risks. Furthermore, the successful implementation of preprocessing, model training, and evaluation procedures validates the feasibility of applying machine learning techniques for structured student category prediction in school environments.

```
dt_model = Pipeline(steps=[
    ("preprocess", preprocessor),
    ("model", DecisionTreeClassifier(random_state=42))
])
```

Figure 3. Decision Tree Model Pipeline

```
rf_model = Pipeline(steps=[
    ("preprocess", preprocessor),
    ("model", RandomForestClassifier(
        n_estimators=100,
        random_state=42
    ))
])
```

Figure 4. Random Forest Model Pipeline

```
xgb_model = Pipeline(steps=[
    ("preprocess", preprocessor),
    ("model", XGBClassifier(
        n_estimators=200,
        learning_rate=0.1,
        max_depth=5,
        subsample=0.9,
        colsample_bytree=0.9,
        random_state=42,
        eval_metric="mlogloss"
    ))
])
```

Figure 5. XGBoost Model Pipeline

```
dt_model.fit(X_train, y_train)
rf_model.fit(X_train, y_train)
xgb_model.fit(X_train, y_train)
```

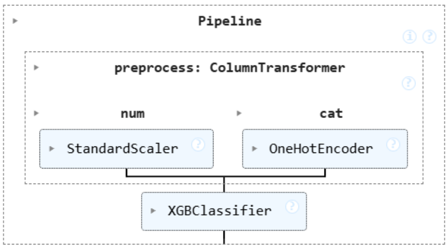

Figure 6. Model Pipelines w/transformer

```
def evaluate_model(name, model, X_test, y_test):
    y_pred = model.predict(X_test)

    acc = accuracy_score(y_test, y_pred)
    prec = precision_score(y_test, y_pred, average="weighted", zero_division=0)
    rec = recall_score(y_test, y_pred, average="weighted", zero_division=0)
    f1 = f1_score(y_test, y_pred, average="weighted", zero_division=0)

    print(f"=== {name} ===")
    print(f"Accuracy : {acc:.4f}")
    print(f"Precision: {prec:.4f}")
    print(f"Recall   : {rec:.4f}")
    print(f"F1-score : {f1:.4f}")
    print("\nClassification report:")
    print(classification_report(y_test, y_pred, zero_division=0))
    print("-" * 40)
```

Figure 7. Machine Learning Model Evaluation

Despite the promising results, this study acknowledges several limitations. The dataset, although representative in structure, remains limited in size and diversity. Additionally, only baseline models were tested, and no hyperparameter optimization or advanced feature engineering techniques were applied. Future research should incorporate longitudinal data, larger and more varied student samples, and more sophisticated machine learning models to enhance predictive accuracy and model generalization. Nonetheless, the findings contribute meaningfully to the ongoing development of practical, reproducible, and accessible ML frameworks for smaller educational institutions.

V. Conclusion

This study evaluated the effectiveness of several machine learning algorithms for predicting student performance categories as a foundation for personalized curriculum design. Using a dataset consisting of 120 student records from Independent School Batam, three models, which are Decision Tree, Random Forest, and XGBoost were implemented and compared using evaluation metrics including accuracy, precision, recall, and F1-score. The experimental results indicate that the Random Forest model achieved the best overall performance, with an accuracy of 0.83 and an F1-score of 0.82. These findings demonstrate that ensemble-based learning methods are effective in capturing complex relationships among student-related variables such as academic performance, attendance, assignment completion, and participation levels.

Further analysis using the confusion matrix revealed that the Random Forest model performed particularly well in identifying students in the higher performance category, correctly classifying all instances in that class. Some misclassifications were observed between the medium and higher performance categories, which may be attributed to similarities in learning behavior and academic indicators among these groups. Nevertheless, the overall classification performance suggests that machine learning techniques can provide meaningful insights into student learning patterns.

From a practical perspective, the results highlight the potential of machine learning models to support data-driven decision-making in educational environments. By identifying patterns associated with student performance, schools can use predictive insights to design more personalized learning strategies and targeted academic interventions. Such approaches may help educators better support students with varying learning needs and improve overall educational outcomes.

Despite these promising results, this study is subject to certain limitations. The dataset used in the experiments contains only 120 student records, which may limit the generalizability of the findings to larger educational settings. Future research could expand the dataset to include more students, incorporate longitudinal academic data, and explore additional machine learning techniques to further improve predictive performance and practical applicability in educational analytics.

Acknowledgment

The author would like to express sincere gratitude to the School of Computer Science, Bina Nusantara University, for providing the academic support and facilities necessary to complete this research. Appreciation is also extended to the faculty members and advisors whose guidance, feedback, and encouragement greatly contributed to the development and refinement of this study.

Special thanks are given to the reviewers who provided valuable insights and constructive comments on earlier drafts, helping to strengthen the clarity, methodology, and academic rigor of this work. The author also wishes to acknowledge the contributions of peers and colleagues who offered technical assistance and shared their expertise during the data processing and model implementation stages.

Although no external funding was received for this research, the institutional environment and academic resources offered by the university played an essential role in supporting the successful completion of this study.

REFERENCES

[1] M. Yanto, Mad Sa'I, and Nailatur Rizqiyah, "Personalisasi Pendidikan Berbasis AI dalam Meningkatkan Kualitas Belajar Siswa," *Entita: Jurnal Pendidikan Ilmu Pengetahuan Sosial dan Ilmu-Ilmu Sosial*, pp. 507–522, May 2025, doi: 10.19105/ejpis.v1i.19116.

[2] J. Putri and A. Yudiawan, "ANALISIS KEMAMPUAN GURU DALAM MENGINTEGRASIKAN MATERI AJAR DENGAN KEBUTUHAN BELAJAR PESERTA DIDIK DI SEKOLAH DASAR," *Jurnal Ilmu Pendidikan (JIP)*, vol. 3, no. 3, pp. 491–502, 2025.

[3] E. Rumahlewang, M. Amin Lasaiba, L. Lokollo, and F. S. Noya, "EFEKTIVITAS E-LEARNING DAN KOMPETENSI ASESMEN DALAM MENINGKATKAN PRESTASI AKADEMIK," *SAP (Susunan Artikel Pendidikan*, vol. 9, no. 3, 2025.

[4] S. H. Soro, E. Hayati, D. Tejawati3, and A. Susanti4, "Optimalisasi Perencanaan Strategik Pembelajaran Dalam Meningkatkan Mutu Lulusan Era Digital," *EDUKASIA: Jurnal Pendidikan dan Pembelajaran*, vol. 5, pp. 2243–2252, 2024, [Online]. Available: https://jurnaledukasia.org

[5] F. Fithroni, W. Ika Setiyati, W. Aditiono, and R. Krismanto Priyoatmojo, "Langkah Kolaboratif Guru dalam Manajemen dan Intervensi Pembelajaran Model Pull Out bagi Siswa Slow Learner," *Ideguru : Jurnal Karya Ilmiah Guru*, vol. 9, no. 2, 2024, doi: 10.51169/ideguru.v9i2.981.

[6] E. Kalita *et al.*, "Educational data mining: a 10-year review," Dec. 01, 2025, *Springer Science and Business Media B.V.* doi: 10.1007/s10791-025-09589-z.

[7] H. Al Aziz and H. A. Santoso, "Model Prediksi Stunting Anak di Indonesia Menggunakan Extreme Gradient Boosting," *Jurnal Algoritma*, 2025, doi: 10.33364/algoritma/v.22-1.2289.

[8] M. R. Islam, A. M. Nitu, M. A. Marjan, M. P. Uddin, M. I. Afjal, and M. A. Al Mamun, "Enhancing tertiary students' programming skills with an explainable Educational Data Mining approach," *PLoS One*, vol. 19, no. 9 September, Sep. 2024, doi: 10.1371/journal.pone.0307536.

[9] P. R. Pitchapati, S. R. Vemula, and M. C. Moraes, "Evaluating Machine Learning Algorithms for Student Performance Prediction in Real-Time Learning Analytics Dashboards," in *Proceedings of the 57th ACM Technical Symposium on Computer Science Education V.2*, New York, NY, USA: ACM, Feb. 2026, pp. 1481–1482. doi: 10.1145/3770761.3777204.

[10] N. Shirahama, "A Quasi-Experimental Study of Parallel Python-R Learning in Data Science Education: Implementation and Assessment in a BYOD Environment," 2025.

[11] M. Rahbari, S. Rahlfs, E. Jortzik, I. Bogeski, and K. Becker, "H2O2 dynamics in the malaria parasite Plasmodium falciparum," *PLoS One*, vol. 12, no. 4, Mar. 2017, doi: 10.1371/journal.

[12] A. Ariyanti, Y. T. Herlambang, and T. Muhtar, "Urgensi Kompetensi Pedagogik Guru dalam Pembelajaran Abad Ke- 21: Studi Kritis Pedagogik Futuristik," *Ideguru: Jurnal Karya Ilmiah Guru*, vol. 10, no. 1, pp. 389–395, Nov. 2024, doi: 10.51169/ideguru.v10i1.1417.

[13] I. Datuzuhriah *et al.*, "TEORI KONSTRUKTIVISTIK DAN IMPLIKASINYA DALAM PEMBELAJARAN PENDIDIKAN AGAMA ISLAM DI SMPN 1 RAMBUTAN," *Jurnal Pendidikan Agama Islam*, vol. 12, 2025.

[14] N. Nafakoti and Sri Atun, "Pengaruh Inquiry-based Contextual Learning terhadap Kemampuan Berpikir Kritis dan Efikasi Diri pada Materi Laju Reaksi dalam Menyongsong Pendidikan yang Berkualitas," *Jurnal Pendidikan Matematika dan Sains*, vol. 13, no. Special_issue, pp. 102–114, Aug. 2025, doi: 10.21831/jpms.v13iSpecial_issue.88419.

[15] S. Madani Rambe and S. Yeli, "DIGITAL DIVIDE DALAM PENERAPAN PENDEKATAN DEEP LEARNING DI SEKOLAH NEGERI KOTA PEKANBARU," *Research and Development Journal Of Education*, vol. 11, no. 2, pp. 942–952, 2025, doi: 10.30998/rdje.v11i2.23989.

[16] T. J. Sinaga, "Jurnal J-MendiKKom (Jurnal Manajemen, Pendidikan dan Ilmu Komputer) Analitik Pendidikan 4.0: Penerapan Data Mining dalam Mengungkap Karakteristik Siswa," *Jurnal J-MendiKKom (Jurnal Manajemen, Pendidikan dan Ilmu Komputer)*, vol. 2, no. 2, pp. 3046–5893, 2025.

[17] V. M. Dinata, A. Wedi, and O. Fajarianto, "Pengembangan Model Pembelajaran Adaptif dengan Implementasi Learning Analytics Berdasarkan Gaya Belajar VARK," 2024. [Online]. Available: https://jurnaldidaktika.org

[18] P. Chyan *et al.*, *Pengantar Data Science Mengambil Keputusan Berdasarkan Data PT. MIFANDI MANDIRI DIGITAL*. 2024.

[19] D. Anggraini, "DATA MINING PENDIDIKAN: PREDIKSI GAYA BELAJAR MAHASISWA TEKNIK MENGGUNAKAN MACHINE LEARNING," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 11, no. 3, pp. 563–572, 2025.

[20] M. Ardiansyah and M. L. Nugraha, "IMPLEMENTASI DEEP LEARNING UNTUK MENINGKATKAN HASIL PEMBELAJARAN DI SEKOLAH MENENGAH KEJURUAN (SMK) SE-JAKARTA BARAT," *Research and Development Journal of Education*, vol. 11, no. 1, p. 302, Apr. 2025, doi: 10.30998/rdje.v11i1.26453.

[21] H. A. Nabila and Endang Wahyu Pamungkas, "PERBANDINGAN ALGORITMA MACHINE LEARNING: SVM, RANDOM FOREST, DAN XGBOOST UNTUK PREDIKSI STROKE," *Rabit : Jurnal Teknologi dan Sistem Informasi Univrab*, vol. 10, no. 2, pp. 1098–1110, Jul. 2025, doi: 10.36341/rabit.v10i2.6444.

[22] A. Kurniawan and R. Abdul Aziz, "Prediksi Kabut Bandar Udara di Indonesia Menggunakan Neural Network dan Radom Forest," *Building of Informatics, Technology and Science (BITS)*, vol. 6,

no. 2, pp. 746–757, Sep. 2024, doi: 10.47065/bits.v6i2.5544.
[23] E. D. Anggara, A. Widjaja, and B. R. Suteja, "Prediksi Kinerja Pegawai sebagai Rekomendasi Kenaikan Golongan dengan Metode Decision Tree dan Regresi Logistik," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, no. 1, Apr. 2022, doi: 10.28932/jutisi.v8i1.4479.
[24] R. Irfannandhy, L. B. Handoko, and N. Ariyanto, "Analisis Performa Model Random Forest dan CatBoost dengan Teknik SMOTE dalam Prediksi Risiko Diabetes," *Edumatic: Jurnal Pendidikan Informatika*, vol. 8, no. 2, pp. 714–723, Dec. 2024, doi: 10.29408/edumatic.v8i2.27990.
[25] A. P. Sari, Billy, D. A. Tsaqif, B. Sartono, and A. R. Firdawanti, "Classification of Drinking Water Source Suitability in West Java Using XGBoost and Cluster Analysis Based on SHAP Values," *Indonesian Journal of Statistics and Its Applications*, vol. 8, no. 2, pp. 202–214, Dec. 2024, doi: 10.29244/ijsa.v8i2p202-214.
[26] S. Mujiyono, U. P. Sanjaya, I. S. Wibisono, and H. Setyowati, "Prediksi Fluktuasi Berat Badan Berdasarkan Pola Hidup Menggunakan Model XGBoost dan Deep Learning," *Jurnal Algoritma*, vol. 22, no. 1, pp. 221–233, May 2025, doi: 10.33364/algoritma/v.22-1.2253.
[27] S. N. Safitri, H. Setiadi, and E. Suryani, "Educational Data Mining Using Cluster Analysis Methods and Decision Trees based on Log Mining," *Jurnal RESTI*, vol. 6, no. 3, pp. 448–456, Jun. 2022, doi: 10.29207/resti.v6i3.3935.
[28] A. Wibowo, M. Kom, and M. Si, *Pengantar AI, BIG DATA dan ILMU DATA*. 2025.
[29] S. Rohmah Nurpadilah and K. Nur Amany, "HukumMeminjam Uang Di Pegadaian Syariah Dengan Menggunakan Data-Data Milik Orang Lain," *Jurnal Kajian Islam Dan Sosial Keagamaan*, vol. 2, no. 3, p. 429, 2025, [Online]. Available: https://jurnal.ittc.web.id/index.php/jkis/index
[30] Y. Shi, F. Sun, H. Zuo, and F. Peng, "Analysis of Learning Behavior Characteristics and Prediction of Learning Effect for Improving College Students' Information Literacy Based on Machine Learning," *IEEE Access*, vol. 11, pp. 50447–50461, 2023, doi: 10.1109/ACCESS.2023.3278370.
[31] S. Gangadhar Patchipala, "Data Anonymization in AI and ML Engineering: Balancing Privacy and Model Performance Using Presidio," 2023.
[32] R. Fitriansyah and R. Tommy Gumelar, "Prediksi Faktor Risiko Gangguan Tidur Menggunakan Pendekatan Machine Learning Logistic Regression Dan Gradient Boosting," vol. 5, no. 2, p. 2025, 2025.
[33] Y. Gong, G. Liu, Y. Xue, R. Li, and L. Meng, "A survey on dataset quality in machine learning," *Inf. Softw. Technol.*, vol. 162, Oct. 2023, doi: 10.1016/j.infsof.2023.107268.
[34] P.-O. Côté, A. Nikanjam, N. Ahmed, D. Humeniuk, and F. Khomh, "Data Cleaning and Machine Learning: A Systematic Literature Review," May 2024, [Online]. Available: http://arxiv.org/abs/2310.01765
[35] H. A. Ahmed, P. J. Muhammad Ali, A. K. Faeq, and S. M. Abdullah, "An Investigation on Disparity Responds of Machine Learning Algorithms to Data Normalization Method," *ARO-The Scientific Journal of Koya University*, vol. 10, no. 2, pp. 29–37, Dec. 2022, doi: 10.14500/aro.10970.
[36] A. Decoux *et al.*, "Comparative performances of machine learning algorithms in radiomics and impacting factors," *Sci. Rep.*, vol. 13, no. 1, Dec. 2023, doi: 10.1038/s41598-023-39738-7.
[37] T. Gori, A. Sunyoto, and H. Al Fatta, "Preprocessing Data dan Klasifikasi untuk Prediksi Kinerja Akademik Siswa," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 11, no. 1, pp. 215–224, Feb. 2024, doi: 10.25126/jtiik.20241118074.
[38] M. Nurus Siroj, A. Khanif Zyen, G. Wahyu, and N. Wibowo, "Optimizing Decision Tree and Random Forest with Grid Search and SMOTE for Malware Classification on IoT Network Traffic," 2025. [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC
[39] E. J. Kusuma, R. Nurmandhani, L. Aryani, I. Pantiawati, and G. F. Shidik, "Optimasi Model Extreme Gradient Boosting Dalam Upaya Penentuan Tingkat Risiko Pada Ibu Hamil Berbasis Bayesian Optimization (BOXGB)," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 12, no. 1, pp. 111–120, Feb. 2025, doi: 10.25126/jtiik.2025129001.
[40] A. P. Rahmadina, N. Y. Setiawan, and A. N. Rusydi, "ANALISIS SENTIMEN TERHADAP KEBIJAKAN KENAIKAN PPN 12% MENGGUNAKAN SUPPORT VECTOR MACHINE (SVM)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 9, no. 9, pp. 2548–964, 2025, [Online]. Available: http://j-ptiik.ub.ac.id
[41] I. Gede, J. Kurniarwan, C. Dewi, and M. A. Rahman, "PENERAPAN MACHINE LEARNING EXTREME GRADIENT BOOSTING DALAM KLASIFIKASI POTENSI TSUNAMI BERDASARKAN DATA GEMPA BUMI," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 9, no. 2, pp. 2548–964, 2025, [Online]. Available: http://j-ptiik.ub.ac.id
[42] M. J. Faisti, R. Hadapiningradja Kusumodestoni, G. Wahyu, and N. Wibowo, "Mental Health Classification Using Naïve Bayes and Random Forest Algorithms," 2025. [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC