PERFECT
JOURNAL OF SMART ALGORITHMS

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

Prediction Of Repeating Object-Oriented Programming Course for Informatics Students at ITAF Kupang Using Extreme Gradient Boosting (XGBoost)

Mohamad Iqbal Ulumando
Informatics Study Program, Institut Teknologi Alberth Foenay Kupang, Indonesia
iqbal77ulumando@gmail.com

open access

***Corresponding Author**

Article History:
Submitted: 26-05-2026
Accepted: 07-06-2026
Published: 12-06-2026

Keywords:
XGBoost; Student Prediction; Learning Behavior; Object Oriented Programming; Machine Learning.

ABSTRACT

The Object-Oriented Programming course is one of the core courses in the Informatics Study Program which has a fairly high level of difficulty so that some students have the potential to fail and have to repeat the course. This study aims to build a prediction model for students of the Informatics Study Program at ITAF Kupang who have the potential to repeat the Object-Oriented Programming course using the Extreme Gradient Boosting (XGBoost) algorithm based on student learning behavior data. The data used amounted to 60 students with variables including attendance, assignment grades, accuracy of assignment submission, discussion participation, quiz scores, practicum activities, and mid-term/final exam scores. The research stages include data collection, data preprocessing, training and testing data distribution, XGBoost model training, and model evaluation using Confusion Matrix, Accuracy, Precision, Recall, and F1-Score. The results of the study showed that the XGBoost model was able to perform good classification with an Accuracy value of 83.33%, Precision of 80.00%, Recall of 80.00%, and F1-Score of 80.00%. Feature importance analysis showed that quiz scores were the most influential factor in students' potential to repeat courses, followed by mid-term/final exam scores and assignment scores. The results of the study proved that student learning behavior data can be used to build an early warning system that helps lecturers and study programs identify at-risk students early on so that more effective academic mentoring can be provided.

INTRODUCTION

Advances in information technology have driven the use of data in various fields, including higher education. Universities now generate a variety of academic data that can be used to support decision-making and improve the quality of learning. One way to utilize this data is through the application of machine learning to identify students at risk of academic difficulties so that early intervention can be provided (Sulehu et al., 2025).

In the ITAF Kupang Informatics Study Program, the Object-Oriented Programming (OOP) course is one of the core courses that plays an important role in building student competencies in the field of software development (Farhan et al., 2025). However, students often find the material covering the concepts of classes, objects, encapsulation, inheritance, and polymorphism quite complex. This leads to some students achieving less than optimal learning outcomes and potentially having to repeat the course in the following semester (Lubis & Putri, 2025).

Until now, identifying students who are likely to repeat a course has generally been done based on exam scores or final year results. This approach tends to be delayed because lecturers only learn about a student's condition after the learning process is nearly complete (Kurniasih, 2024). In fact, learning behavior data such as attendance levels, accuracy of assignment submission, participation in discussions, practical activities, and quiz results can provide an initial picture of the level of student involvement and readiness to attend lectures (Sulehu et al., 2025).

One of the machine learning methods that is widely used for classification tasks is Extreme Gradient Boosting (XGBoost) (Dachi, 2023). This algorithm is a development of the gradient boosting method which has high capability in producing accurate predictions, handling complex data, and reducing the risk of overfitting (Murdiansyah, 2024). In addition, XGBoost is able to identify the features that have the most influence on the prediction results so that it can help understand the factors that cause students to potentially repeat courses (Sari & Budiarni, 2025).

Based on these problems, this study aims to develop a prediction model for ITAF Kupang Informatics Study Program students who have the potential to repeat the Object-Oriented Programming course using the XGBoost algorithm based on learning behavior data. The results of this study are expected to support the development of an early warning system for more effective academic mentoring (Azizah et al., 2026).

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

LITERATURE REVIEW

Student Academic Predictions

Academic prediction is the process of estimating a student's academic performance or achievement based on available data. In higher education, academic prediction is used to identify students at risk of declining academic performance, delays in their studies, or failure to complete certain courses (Putra & Harahap, 2024). The use of data analysis and machine learning techniques enables educational institutions to detect students who need academic assistance early, thereby increasing the level of learning success (Sitanggang et al., 2026).

Student Learning Behavior

Learning behavior is a series of activities undertaken by students during the learning process that reflect their level of engagement and commitment to attending lectures. Some indicators of learning behavior commonly used in educational research include attendance, participation in discussions, timely submission of assignments, practical activities, frequency of access to learning materials, and results of periodic evaluations (Harkamsyah Andrianof et al., 2025). Good learning behavior is generally positively correlated with students' academic achievement, so that learning behavior data can be used as a predictor variable in building an academic prediction model, such as previous research conducted by the author (Ulumando, 2025).

Machine Learning in Education

Machine learning is a branch of artificial intelligence (AI) that enables computer systems to learn patterns from data and generate predictions without explicit programming. In education, machine learning is widely used to predict student academic performance, identify dropout risks, analyze learning patterns, and support academic decision-making (Sulehu et al., 2025). Commonly used algorithms include Decision Tree, Random Forest, Naive Bayes, Support Vector Machine (SVM), Artificial Neural Network, and XGBoost. Choosing the right algorithm significantly impacts the accuracy and quality of the resulting predictions (Lubis & Putri, 2025).

Extreme Gradient Boosting (XGBoost)

Extreme Gradient Boosting (XGBoost) is an ensemble learning algorithm developed from the Gradient Boosting Decision Tree (GBDT) method. XGBoost works by gradually building a number of decision trees, where each new tree serves to correct the prediction errors of the previous tree (Hayatunnisa et al., 2025). This algorithm has several advantages, such as high computing speed, the ability to handle large amounts of data, regularization features to reduce overfitting, and the ability to measure the level of importance of each feature (feature importance) (Hidayat et al., 2026).

In general, the objective function in XGBoost can be expressed as:

$$Obj = \sum_{i=1}^{n} l\,(y_1, \hat{y}_i) + \sum_{k=1}^{K} \Omega\,(f_k) \tag{1}$$

Where $l(y_1, \hat{y}_i)$ is a loss function that measures the difference between the actual and predicted values, whereas $\Omega\,(f_k)$ is a regularization function used to control model complexity. With this approach, XGBoost is able to produce models that have a high level of accuracy and are stable in various classification and prediction cases (Efendi & Suharjo, 2026).

Evaluation of Classification Models

Classification model evaluation is performed to measure the model's ability to accurately predict data classes. One commonly used evaluation method is the Confusion Matrix, which consists of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). Based on these values, several evaluation metrics can be calculated, namely Accuracy, Precision, Recall, and F1-Score (Hidayat et al., 2026). The Accuracy formula can be stated as follows:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2}$$

Meanwhile, Precision and Recall are calculated using the equation:

$$Precision = \frac{TP}{TP+FP} \tag{3}$$

$$Recall = \frac{TP}{TP+FN} \tag{4}$$

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

METHOD

Types of research

This quantitative study uses a machine learning approach to develop a predictive model for students in the Informatics Study Program at ITAF Kupang who are likely to repeat the Object-Oriented Programming (OOP) course. The predictive model was developed using the Extreme Gradient Boosting (XGBoost) algorithm based on student learning behavior data during the course. The model results are used to classify students into categories of those with or without the potential to repeat the course (Izhari, 2025).

Data Sources and Collection

The data used in this study is academic data from students who have taken the Object-Oriented Programming course in the Informatics Study Program at ITAF Kupang. The data was obtained from the academic system, lecturer lecture notes, and learning platforms used during the lecture process, as in previous research conducted by the author (Ulumando, 2026). The variable "Repeat Status" is used as the target variable in the classification process. The variables used in the study include:

Table 1. Research Variable Data

No	Variable	Information
1	Presence	Percentage of student attendance during lectures
2	Assignment Value	Average grades of assignments obtained by students
3	Accuracy of Assignment Collection	Percentage of assignments submitted on time
4	Discussion Participation	Level of student activity in class discussions
5	Quiz Grade	Average student quiz scores
6	Practical Activities	Level of student involvement in practical activities
7	UTS/UAS Value Scores	Midterm Exam Scores
8	Repeating Status	Class label (Yes/No)

Research Stages

The research phase began with data collection on the learning behavior of students who had taken the Object-Oriented Programming course in the Informatics Study Program at ITAF Kupang. The data then underwent preprocessing to ensure the quality of the data used in the study. This process included data cleaning, handling missing values, removing duplicate data, and transforming the data to suit the model's requirements (Ulumando, 2025). Once the data is ready for use, the dataset is divided into training data and testing data. The training data is used to build and train the predictive model, while the testing data is used to measure the model's ability to predict data that has not been previously studied. Next, a model is built using the XGBoost algorithm to classify students who are and are not likely to repeat the Object-Oriented Programming course based on their learning behavior data (Hayuningtyas et al., 2025).

The next stage is model evaluation to determine the level of performance and accuracy of the resulting predictions. The evaluation is conducted using several classification metrics such as Accuracy, Precision, Recall, and F1-Score. After the evaluation process is complete, an analysis of the prediction results and the level of influence of each variable is conducted using the feature importance feature in XGBoost. This analysis aims to identify the learning behavior factors that most contribute to students' potential for repeating courses. This research results can be used as a basis for academic decision-making and the development of an early warning system (Tribuana et al., 2025).

Data collection

The initial stage involved collecting data on the learning behavior of students who had taken the Object-Oriented Programming course. The collected data was then screened to ensure the completeness and consistency of the information required for the analysis process.

Data Preprocessing

This stage prepares the data before it is used in the model training process. The preprocessing stages include:

1. Data cleaning.
2. Handling missing values.
3. Removing duplicate data.
4. Transforming categorical data into numeric data.
5. Normalizing or standardizing the data, if necessary.

The goal of this stage is to produce a high-quality dataset that can improve the performance of the predictive model.

PERFECT JOURNAL OF SMART ALGORITHMS

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

Data Sharing

The processed dataset was divided into training data and testing data. In this study, 80% of the data was used for model training and 20% for model testing. This random split allowed the model to optimally learn data patterns and reduce the potential for bias in the prediction results.

XGBoost Model Development

After the data is divided, the model is trained using the XGBoost algorithm. This algorithm works by gradually building a number of decision trees to minimize prediction errors in previous iterations. The main parameters used in the model include:

1. Number of Estimators (n_estimators)
2. Learning Rate
3. Maximum Depth (max_depth)
4. Subsample
5. Colsample_bytree

The training process is performed using Python software with the XGBoost library.

Model Evaluation

Model performance was evaluated using the Confusion Matrix and several classification evaluation metrics, namely Accuracy, Precision, Recall, and F1-Score.

The Accuracy formula is:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{5}$$

The Precision formula is:

$$Precision = \frac{TP}{TP+ FP} \tag{6}$$

The Recall formula is:

$$Recall = \frac{TP}{TP+FN} \tag{7}$$

The F1-Score formula is:

$$F1 - Score = 2\ x\ \frac{Precision\ x\ Recall}{Precision+ Recall} \tag{8}$$

Description:

1. TP (True Positive): Students who were predicted to repeat and actually did.
2. TN (True Negative): Students who were predicted not to repeat and actually did not.
3. FP (False Positive): Students who were predicted to repeat but actually did not.
4. FN (False Negative): Students who were predicted not to repeat but actually did.

Feature Importance Analysis

In addition to making predictions, this study also analyzed the influence of each variable on the classification results using the Feature Importance feature in XGBoost. This analysis aimed to identify the learning behavior factors that most contribute to students' likelihood of repeating Object-Oriented Programming courses. The results of the Feature Importance analysis can be used by lecturers and study programs as a basis for developing learning strategies and implementing academic interventions for at-risk students.

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

Research Framework

In general, the research flow can be summarized as follows:

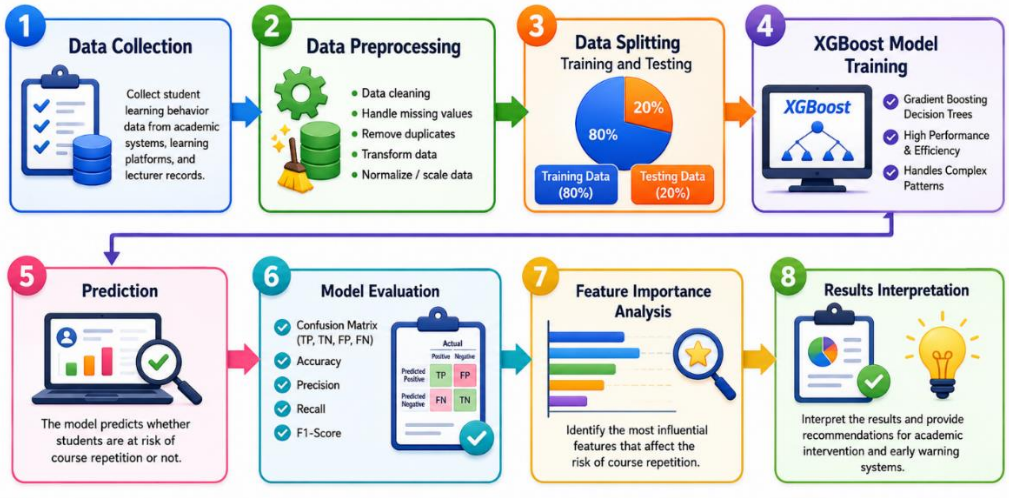

Figure 1. Research flow

Through these stages, it is hoped that a prediction model will be obtained that is able to identify students who have the potential to repeat the Object-Oriented Programming course accurately so that it can support the implementation of an early warning system in the ITAF Kupang Informatics Study Program.

RESULT

Description of Research Data

This study used learning behavior data from 60 Informatics students at the ITAF Kupang Informatics Study Program taking the Object-Oriented Programming course. The variables analyzed included attendance, assignment grades, accuracy of assignment submissions, discussion participation, quiz scores, lab activities, and midterm and final exam scores. This data was used as the basis for developing a prediction model for students who are likely to repeat the course using the XGBoost algorithm.

Student Attendance Rate

Based on the data collection results, an average student attendance rate of 85% was obtained during the Object-Oriented Programming course. This percentage indicates that the majority of students demonstrated a good level of participation in the classroom learning process. High attendance is a crucial indicator of learning success because students have greater opportunities to receive material directly, interact with lecturers, and participate in various learning activities.

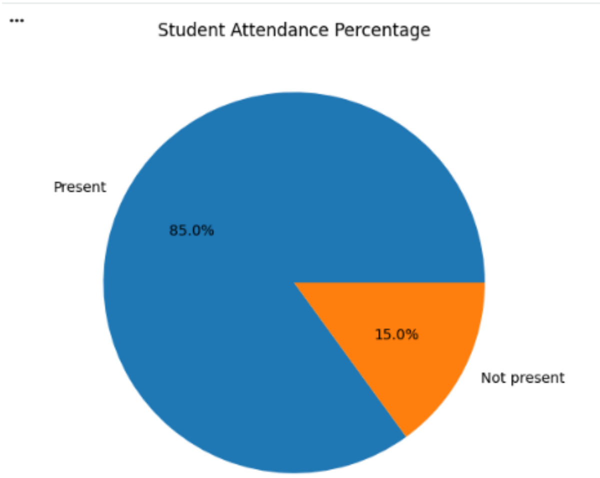

Figure 2. Graph of Student Attendance Level

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

Figure 2 above shows a comparison between the percentage of student attendance and absences over the course of a semester. It can be seen that the proportion of students who attended was significantly greater than the proportion of students who were absent. This condition indicates that students generally have a fairly good commitment to the lecture process. However, there is still an absence rate of approximately 15%, which has the potential to impact student understanding of the material, especially in the Object-Oriented Programming course, which requires continuous practice and conceptual understanding.

Student Assignment Grades

Assignments are one of the evaluation components used to measure students' understanding of the material presented. Based on the analysis, approximately 97% of students completed the assignments given during the lecture. This high completion rate indicates that the majority of students are well engaged in the learning process.

Table 2. Distribution of Student Assignment Grades

Grade	Number of Students
95	12
90	7
80	21
65-67	15
0-50	5

After the assignments were submitted and graded, the score distribution was as shown in Table 2 above. Twelve students received a score of 95, seven students received a score of 90, and 21 students received a score of 80. Thus, 40 students received scores above 80, indicating good academic ability in completing the assigned assignments. On the other hand, 15 students received scores in the 65–67 range, and five students received scores between 0–50. This group of students demonstrated difficulty understanding the material or suboptimal completion of assignments. This situation requires attention as it could be an early indicator of students potentially experiencing academic difficulties in the subsequent evaluation phase.

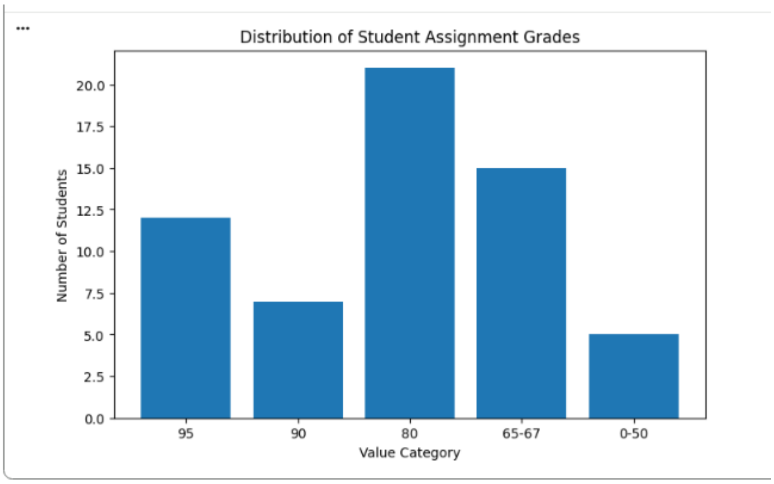

Figure 3. Assignment Value Distribution Graph

Based on Figure 3 above, it can be seen that the 80-point category has the largest number of students compared to other categories. The graph also shows that the distribution of scores tends to be concentrated in the middle to high categories. This indicates that most students are capable of completing the assignment well. However, the presence of students in the low-point category indicates significant variation in ability, requiring special attention to this group of students.

Accuracy of Assignment Collection

The accuracy of assignment submissions is an indicator of student discipline and responsibility during the learning process. Research shows that approximately 70% of students submit assignments on time, while 30% submit them late or even at all.

PERFECT JOURNAL OF SMART ALGORITHMS

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

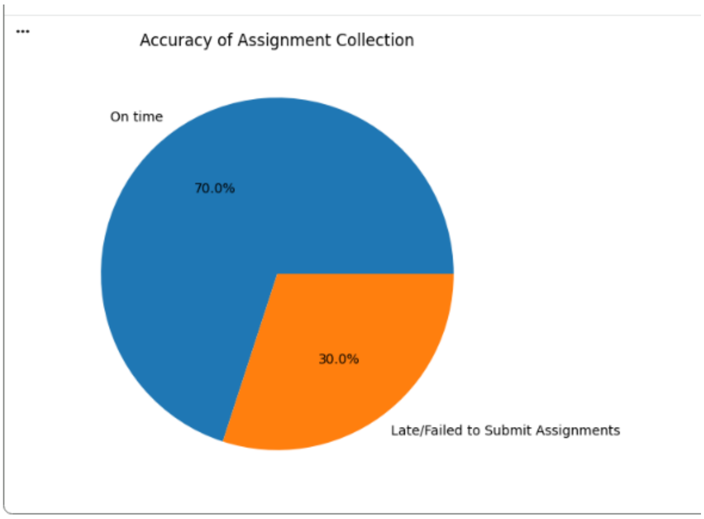

Figure 4. Graph of Accuracy in Collecting Tasks

Figure 4 above shows that the majority of students demonstrate a high level of discipline in meeting specified deadlines. The percentage of students who submit assignments on time is significantly higher than the group of students who submit assignments late or at all. However, a proportion of 30% is still quite large and can be a factor affecting students' overall academic performance. Late assignment submissions are often related to low learning motivation, poor time management, or difficulty understanding the material.

Discussion Participation

Discussion participation is one indicator of student active involvement in the learning process. In Object-Oriented Programming courses, discussions are crucial because they help students understand abstract and complex programming concepts.

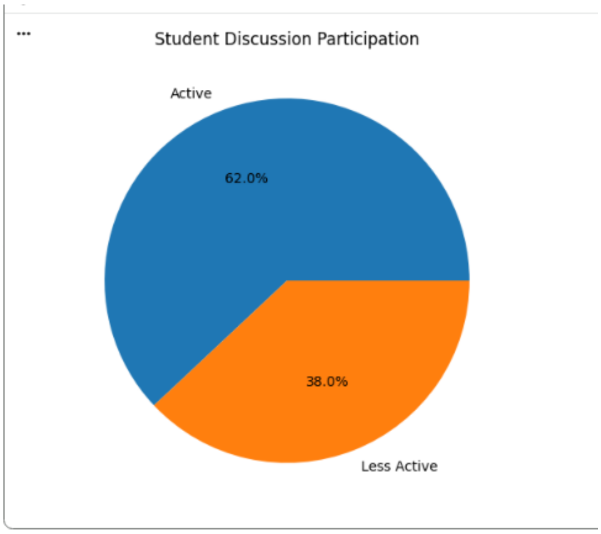

Figure 5. Student Discussion Participation Graph

The results of the study showed that approximately 62% of students were active in class discussions, while 38% were less active. Figure 5 above shows that the number of students actively participating in discussions is greater than the number of students who are passive. This condition indicates that most students have a good level of interest and engagement during the learning process. Students who actively discuss generally have a better understanding of concepts because they are directly involved in the process of exchanging ideas, asking questions, and providing solutions to problems discussed in class.

PERFECT JOURNAL OF SMART ALGORITHMS
E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

Quiz Grade

Quizzes are used as an evaluation tool to measure students' understanding of the material presented over a specific period of time. Quiz results can provide an overview of students' understanding of the material before facing a more serious exam.

Table 3. Distribution of Quiz Scores

Value Range	Number of Students
70-95	30
60-67	15
50-57	7
20-40	5
0	3

Based on Table 3 above, 30 students obtained scores between 70 and 95, indicating a good level of understanding of the material. Fifteen students obtained scores in the 60–67 range, and seven students obtained scores between 50 and 57. Furthermore, five students obtained scores between 20 and 40, and three students received a score of 0 for not taking the quiz. This distribution shows a clear difference in student understanding levels. Some students were able to grasp the material well, while others still struggled to grasp the concepts taught.

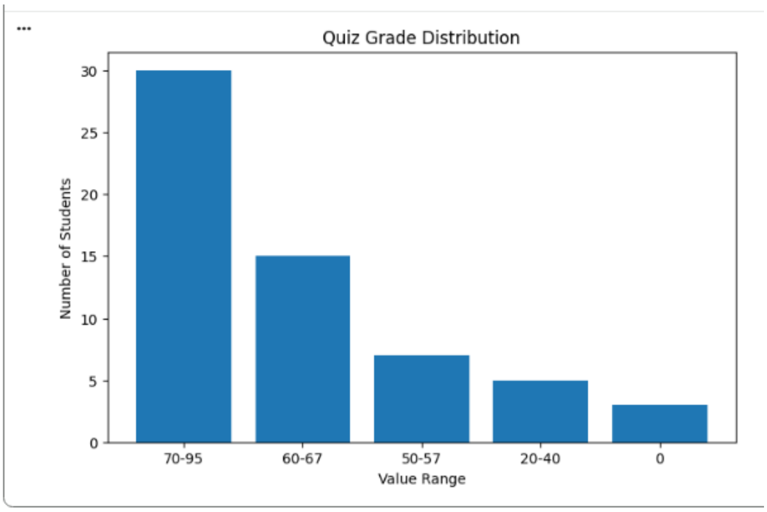

Figure 6. Quiz Score Distribution Graph

Figure 6 above shows that the 70–95 score category dominates student quiz results. However, there were still a number of students who obtained low scores or did not take the quiz. This situation suggests that quiz scores can be an important indicator in identifying students who are potentially experiencing academic difficulties in Object-Oriented Programming courses.

Practical Activities

Practicum is a major component in learning Object-Oriented Programming because students not only understand the theory, but also apply programming concepts directly through practical activities.

Table 4. Practical Activities

Category	Percentage
Active	75 %
Quite Active	20 %
Not active	5 %

PERFECT JOURNAL OF SMART ALGORITHMS
E-ISSN: E-ISSN: 3064-0377
Volume 3, Number 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

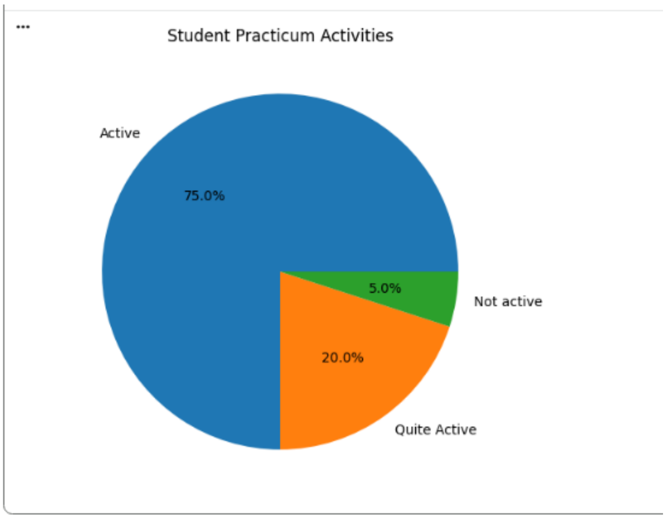

Figure 7. Graph of Student Practical Activities

Based on the research results, it was found that 75% of students actively participated in the practicum, 20% were moderately active, and 5% were inactive. Figure 7 above shows that the majority of students were highly involved in the practicum activities. High levels of practicum activity indicate that students have sufficient opportunities to develop programming skills directly. Conversely, students who are less active or inactive in the practicum may experience difficulties in understanding the implementation of programming concepts learned in class.

UTS/UAS Value Scores

Mid-term and final exam scores are used as indicators of student academic achievement after participating in the learning process for one semester. The results of the study indicate that the distribution of mid-term and final exam scores follows a relatively similar pattern to the distribution of assignment scores.

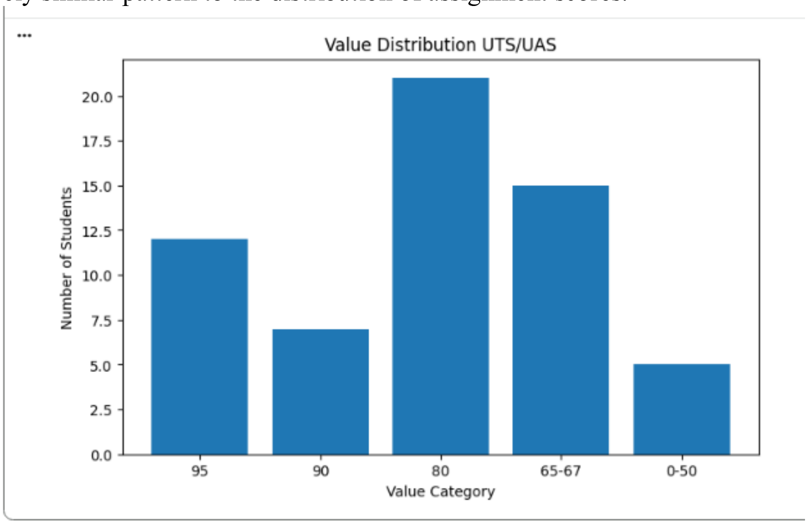

Figure 8. Value Distribution Graph UTS/UAS

Based on Figure 8 above, it can be seen that the majority of students obtained grades in the good to excellent category. This distribution pattern, similar to assignment grades, indicates a strong relationship between students' ability to complete assignments and their success on exams. Students who consistently obtained high assignment grades tended to achieve good mid-term and final exam results. This finding suggests that assignment activity can be an early indicator for predicting student academic success in Object-Oriented Programming courses.

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

Prediction Results Using XGBoost

The dataset was divided into:

1. Training Data = 48 students (80%)
2. Testing Data = 12 students (20%)

The XGBoost model was then trained using student learning behavior variables and produced a classification of students who were likely to repeat the course.

Model Evaluation

The results of the model testing produced the following Confusion Matrix:

Table 5. Confusion Matrix Model Test Results

Model Testing	Repeat Prediction	Prediction Does Not Repeat
Actual Repeat	4	1
Actual Does Not Repeat	1	6

Obtained:

1. TP = 4
2. TN = 6
3. FP = 1
4. FN = 1

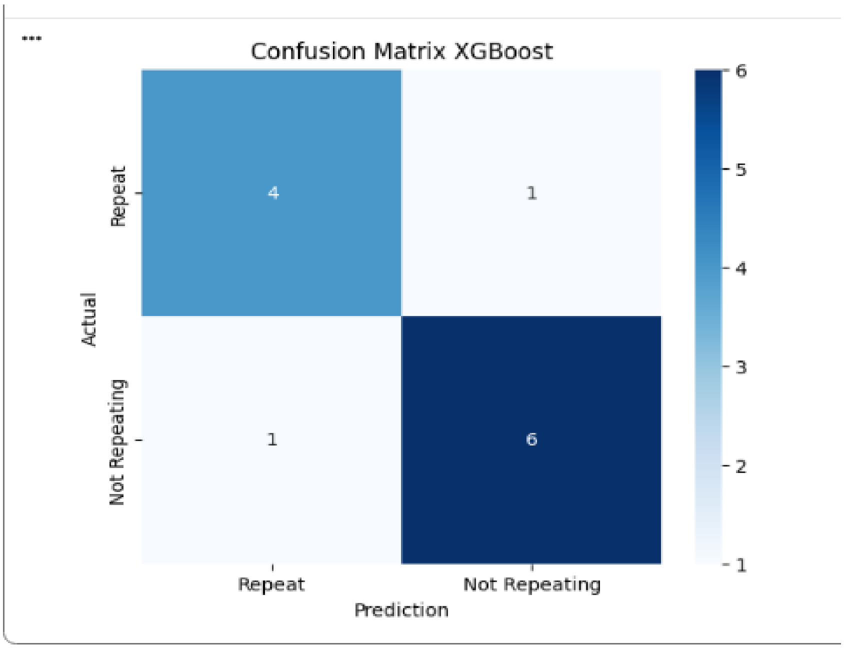

Figure 9. Confusion Matrix Graph

Feature Importance

One of the advantages of the XGBoost algorithm is its ability to identify the feature importance of each variable used in the prediction process. This analysis is conducted to determine which variables are most influential in determining whether a student is likely to repeat an Object-Oriented Programming course. By identifying the most dominant variables, lecturers and study programs can better focus their academic interventions for at-risk students. Based on the results of the XGBoost model training, the feature importance values were obtained as shown in Table 6 below.

PERFECT JOURNAL OF SMART ALGORITHMS

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

Table 6. XGBoost Feature Importance Results

Variable	Importance
Quiz Grade	0.27
UTS/UAS Value Scores	0.23
Assignment Value	0.18
Practical Activities	0.12
Presence	0.09
Accuracy of Assignment Collection	0.07
Discussion Participation	0.04

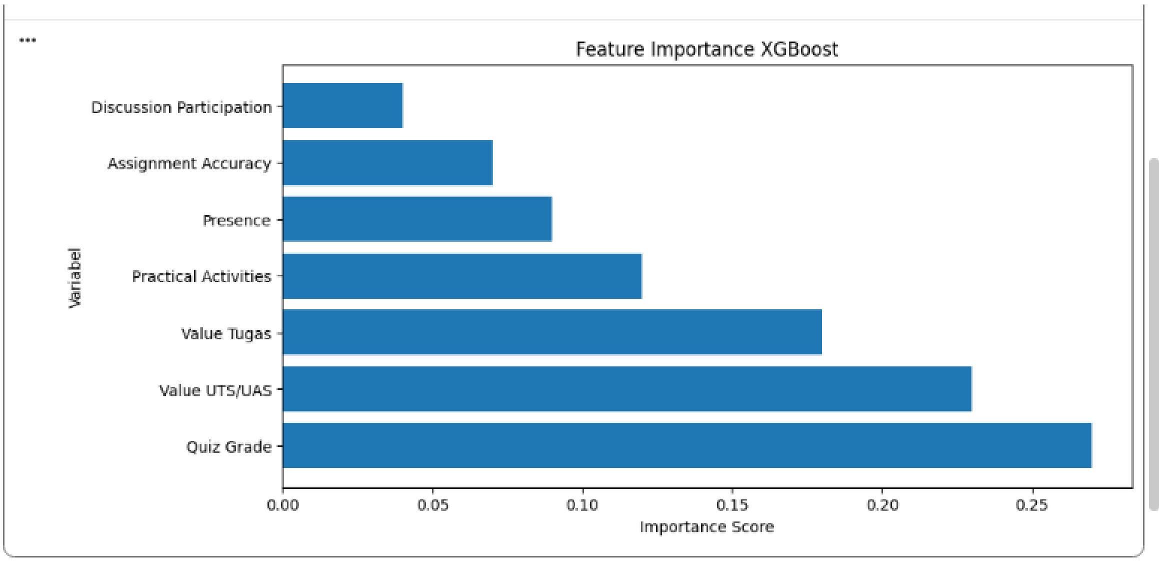

Figure 10. XGBoost Feature Importance Graph

Based on Table 6 and Figure 10, the Quiz Score variable has the highest importance score of 0.27, or 27%. This result indicates that quiz scores are the most influential factor in determining students' likelihood of repeating the Object-Oriented Programming course. This is understandable because quizzes are administered periodically throughout the learning process, thus reflecting students' level of understanding of the material taught. Students with low quiz scores tend to have difficulty grasping the basic concepts that will be used in subsequent materials. The second variable with a significant influence is the Mid-Term/Ultimate Exam Score, with an importance score of 0.23. This value indicates that mid-term and final evaluation results significantly contribute to the classification process. Students with low mid-term or final exam scores generally have a higher risk of repeating the course, as exam results reflect overall mastery of the material.

Next, Assignment Grades ranked third with an importance score of 0.18. These results indicate that students' ability to complete assignments given during lectures is also a crucial factor in determining academic success. Assignments in Object-Oriented Programming courses generally involve problem-solving exercises and implementation of programming concepts, thus demonstrating students' level of understanding of the material being studied. The Practicum Activity variable had an importance score of 0.12. Although its contribution was smaller than quiz and exam scores, practicum activities still had a significant impact on the predicted results. This indicates that student involvement in programming practice activities plays a role in improving their ability to understand and apply Object-Oriented Programming concepts.

Meanwhile, the Attendance variable received an importance score of 0.09. These results indicate that student attendance during lectures is related to academic success, although the effect is not as strong as variables directly related to academic achievement. High attendance provides students with the opportunity to directly participate in material explanations, discussions, and other learning activities. The Assignment Submission and Discussion Participation variables received importance scores of 0.07 and 0.04, respectively. These two variables contributed relatively less than the other variables. However, their presence still provided additional information that helped the model distinguish students who were likely to repeat the course from those who were not. Overall, the results of the feature importance analysis indicated that factors directly related to student academic performance—quiz scores, midterm/final exam scores, and assignment scores—were the most dominant variables in determining students' likelihood of repeating the

PERFECT JOURNAL OF SMART ALGORITHMS

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

Object-Oriented Programming course. This finding indicates that students' ability to understand the material and complete learning evaluations are key indicators that can be used to develop an early warning system. By regularly monitoring quizzes, assignments, and exam scores, instructors can more quickly identify students at risk of academic failure and provide necessary support before the end of the semester.

The results showed that the XGBoost algorithm was able to identify students who were likely to repeat the Object-Oriented Programming course with an accuracy rate of 83.33%. This value indicates that the model has a good ability to distinguish students who are likely to repeat from those who are not. Feature importance analysis showed that quiz scores, mid-term/final exam scores, and assignment scores were the most influential factors in the prediction results. This finding indicates that student academic performance during the learning process is the main indicator in determining the risk of repeating a course. Furthermore, learning behavior variables such as lab activities, attendance, timely assignment submission, and discussion participation also contributed to the model, albeit to a lesser extent. Overall, the results of the study demonstrate that student learning behavior data can be utilized to develop an early warning system that helps lecturers and study programs identify at-risk students early in the semester, allowing for more timely and targeted academic mentoring.

DISCUSSION

The results of the study show that the Extreme Gradient Boosting (XGBoost) algorithm is able to classify students who have the potential to repeat the Object-Oriented Programming (OOP) course with a good level of performance. Accuracy values of 83.33%, Precision of 80.00%, Recall of 80.00%, and F1-Score of 80.00% indicate that the model has a fairly high ability to distinguish students who have the potential to repeat the course from students who can complete the course without repetition. These results indicate that student learning behavior data has a strong relationship with student academic success in the Object-Oriented Programming course. The accuracy rate of 83.33% indicates that the model correctly classified most of the student data. These results support previous studies that have shown XGBoost to be a high-performance machine learning algorithm for classification tasks due to its ability to optimize the learning process through gradual boosting techniques. XGBoost's ability to correct errors at each iteration makes it effective for handling academic data with complex characteristics and many interconnected variables.

Based on the Confusion Matrix results, the model successfully identified 4 students who were truly likely to repeat the course and 6 students who were not. Although there were still one case of False Positive and False Negative each, the number of errors was relatively small compared to the number of correct predictions. In the context of an academic early warning system, the presence of a small number of classification errors is still acceptable because the main goal of the system is to help lecturers identify students who need early attention before they experience academic failure. Thus, the resulting model has the potential to be applied as a decision-making tool in the process of monitoring student learning. The results of the feature importance analysis indicate that quiz scores are the most influential variable in determining students' potential to repeat the course, with an importance value of 0.27 or 27%. This finding indicates that students' ability to complete periodic evaluations during the course is a primary indicator of academic success in the Object-Oriented Programming course. Regular quizzes are able to illustrate students' level of understanding of the material being studied. Students who obtain low quiz scores tend to have difficulty understanding the basic programming concepts that form the foundation for subsequent materials. Therefore, quiz scores can be used as an early indicator to detect students at risk of academic failure.

The second variable with a significant influence was the mid-term and final exam scores, with an importance level of 0.23. These results indicate that formal mid- and final-semester evaluations remain a crucial factor in determining student success. Exam scores reflect a student's ability to comprehensively understand the material and integrate the various concepts learned throughout the semester. This finding reinforces the assumption that students with low exam scores are more likely to repeat courses than those with high scores. Assignment scores ranked third with an importance level of 0.18. This indicates that students' ability to complete assignments given by lecturers significantly contributes to academic success. In the Object-Oriented Programming course, assignments serve not only as an evaluation tool but also as a means of practicing implementing programming concepts. Students who actively work on assignments tend to have a better understanding of the material, thus lowering the risk of repeating the course. Conversely, students who frequently experience difficulty completing assignments indicate obstacles in the learning process that require further attention. The lab activity variable also made a significant contribution, with an importance value of 0.12. These results indicate that student involvement in lab activities has an impact on learning success. Programming is a field that places a strong emphasis on practical skills, so students who actively participate in labs have a greater opportunity to understand the application of concepts they have learned theoretically. These findings suggest that improving the quality and intensity of lab activities can be a strategy to reduce the risk of students repeating courses.

Meanwhile, the attendance variable received an importance score of 0.09. Although its influence is not as significant as other academic variables, attendance remains a contributing factor to student success. High attendance

PERFECT JOURNAL OF SMART ALGORITHMS

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

provides students with the opportunity to directly follow material explanations, interact with lecturers, and gain a more comprehensive learning experience. However, the study results show that attendance alone is not sufficient to guarantee academic success if it is not accompanied by a good understanding of the material, which is reflected in quizzes, assignments, and exam scores. The variables for accuracy of assignment submission and discussion participation have lower importance values than the other variables, namely 0.07 and 0.04, respectively. Although their contributions are relatively small, these two variables still provide additional information that helps the model in classifying. The low importance value for discussion participation may be caused by the presence of students who are active in discussions but do not necessarily have a good understanding of the material, or students who are less active in discussions but still able to achieve high academic results. Therefore, these two variables play a more supporting role than the primary factor in determining the potential for students to repeat courses.

Overall, the results of the study indicate that factors directly related to student academic achievement, namely quiz scores, mid-term/final exam scores, and assignment scores, are the most dominant indicators in predicting students' potential to repeat the Object-Oriented Programming course. This finding suggests that student academic success is more determined by the level of material mastery than simply attendance or participation during the learning process. By utilizing the prediction results generated by the XGBoost model, lecturers and study programs can carry out academic interventions earlier through mentoring activities, tutoring, and regular evaluations of students identified as at risk. From an implementation perspective, the model developed in this study has the potential to be used as a basis for developing an Early Warning System (EWS) in the Informatics Study Program at ITAF Kupang. This system can help lecturers continuously monitor student progress throughout the semester and identify students who need academic assistance before final grades are determined. Thus, the machine learning approach using XGBoost functions not only as a predictive tool but also as a strategic instrument to improve the quality of learning and reduce course repetition rates in higher education.

CONCLUSION

Based on the research results, it can be concluded that the XGBoost algorithm was successfully applied to predict which students in the Informatics Study Program at ITAF Kupang would potentially repeat the Object-Oriented Programming course based on learning behavior data. The model was built using several variables: attendance rate, assignment grades, assignment submission time, discussion participation, quiz scores, lab activities, and mid-term and final exam scores. Test results showed that the model performed well in classification, achieving an accuracy of 83.33%, a precision of 80.00%, a recall of 80.00%, and an F1-score of 80.00%.

The feature importance analysis revealed that quiz scores were the most influential variable in the prediction process, with an importance score of 0.27, followed by mid-term and final exam scores of 0.23, and assignment scores of 0.18. These findings indicate that factors directly related to student academic achievement have a greater influence than other learning behavior factors such as attendance, discussion participation, and assignment submission time.

Overall, this study proves that student learning behavior data can be used to build an effective predictive model in identifying students who have the potential to repeat the Object-Oriented Programming course. The results of this study can be the basis for the ITAF Kupang Informatics Study Program in developing an early warning system to help lecturers and study programs provide academic mentoring more quickly and precisely so that it can increase the level of student learning success.

REFERENCES

Azizah, N. N., Utami, W. D., & Fanani, A. (2026). Algoritma XGBoost Untuk Analisis Prediksi Permintaan Kalibrasi di PT PAL Indonesia. *JIKO (Jurnal Informatika Dan Komputer)*, *10*(1), 14–27. https://doi.org/10.26798/jiko.v10i1.2245

Dachi, J. M. A. S. (2023). Analisis Perbandingan Algoritma XGBoost dan Algoritma Random Forest Ensemble Learning pada Klasifikasi Keputusan Kredit. *Jurnal Riset Rumpun Matematika Dan Ilmu Pengetahuan Alam (JURRIMIPA)*, *2*(2), 87–103.

Efendi, M. A. N., & Suharjo, I. (2026). Perbandingan Algoritma XGBoost dan CatBoost dalam Prediksi Harga Rumah Berdasarkan Data Perumahan di Jabodetabek. *Jurnal Nasional Teknologi Komputer*, *6*(1), 51–63. publikasi.hawari.id/index.php/jnastek

Farhan, A. I., Hatta, A. H., Ramazan, D., Siahaan, F., Anwar, S., & Handono, F. W. (2025). Penerapan Algoritma Long Short Term Memory Untuk Memprediksi Pola Kenaikan Suhu Di Kota Jakarta Pusat. *Jurnal INSAN (Journal of Information Systems Management Innovation)*, *5*(1), 43–52. https://doi.org/10.31294/j-insan.v5i1.9111

Harkamsyah Andrianof, Aggy Pramana Gusman, & Okta Andrica Putra. (2025). Implementasi Algoritma Random Forest untuk Prediksi Kelulusan Mahasiswa Berdasarkan Data Akademik: Studi Kasus di Perguruan Tinggi Indonesia. *Jurnal Sains Informatika Terapan (JSIT)*, *4*(1), 24–28.

PERFECT
JOURNAL OF SMART ALGORITHMS

E-ISSN: E-ISSN: 3064-0377
Volume 3, **Number** 2, July 2026
https://doi.org/10.62671/perfect.v3i2.285

Hayatunnisa, D., Permata, Priandika, A. T., & Gunawan, R. D. (2025). Perbandingan Random Forest dan XGBoost Untuk Prediksi Penjualan Produk E-Commerce Rumah Madu. *Building of Informatics, Technology and Science (BITS)*, *7*(3), 1479–1489. https://doi.org/10.47065/bits.v7i3.8491

Hayuningtyas, R. Y., Yusnaeni, W., Darwati, I., Tania, S., & Abditama, H. (2025). Penerapan Algoritma XGBoost Dalam Menganalisa Keberlanjutan Pelanggan Tour dan Travel. *Evolusi : Jurnal Sains Dan Manajemen*, *13*(2), 25–32. https://doi.org/10.31294/evolusi.v13i2.9748

Hidayat, F. A. K., Sulistianingsih, N., & Hammad, R. (2026). Perbandingan Algoritma XGBoost dan Random Forest dalam Klasifikasi Surat Masuk Pemerintahan. *Journal of Artificial Intelligence and Digital Business (RIGGS)*, *4*(4), 10390–10397. https://journal.ilmudata.co.id/ index.php/RIGGS

Izhari, F. (2025). Pemodelan Perilaku Penggunaan Media Sosial Mahasiswa dengan Algoritma XGBoost. *Jurnal Minfo Polgan*, *14*(1), 1303–1308. https://doi.org/10.33395/jmp.v14i1.15030

Kurniasih, R. (2024). Prediksi Jumlah Mahasiswa Baru Dengan Menggunakan Regresi Linier Sederhana. *Jurnal Matematika*, *24*(1), 84–89. https://journals.unisba.ac.id/index.php/matematika%0ADiterima:

Lubis, A. S., & Putri, R. A. (2025). Analisis Sentimen Mahasiswa Terhadap Penggunaan E-Learning dengan Algoritma Support Vector Machine (SVM). *Decode: Jurnal Pendidikan Teknologi Informasi*, *5*(2), 778–788. https://doi.org/http://dx.doi.org/10.51454/decode.v5i2.1247

Murdiansyah, D. T. (2024). Prediksi Stroke Menggunakan Extreme Gradient Boosting. *JIKO (Jurnal Informatika Dan Komputer)*, *8*(2), 419–426. https://doi.org/10.26798/jiko.v8i2.1295

Putra, M., & Harahap, E. (2024). Machine Learning pada Prediksi Kelulusan Mahasiswa Menggunakan Algoritma Random Forest. *Jurnal Riset Matematika (JRM)*, *4*(2), 127–136. https://doi.org/https://doi.org/10.29313/jrm.v4i2.5102

Sari, F. P., & Budiarni, R. (2025). Implementasi Extreme Gradient Boosting (XGBoost) untuk Klasifikasi Penggunaan Ulang Asisten AI. *Jurnal Pustaka Data*, *5*(2), 303–310. https://doi.org/https://doi.org/10.55382/jurnalpustakadata.v5i2.1402

Sitanggang, A. A., Lase, M. Y., Sipayung, S. P., Informatika, T., Katolik, U., & Thomas, S. (2026). Prediksi Kelulusan Mahasiswa Menggunakan Logistic Regression dan Random Forest Berdasarkan Data Akademik. *Jurnal Pendidikan Tambusai*, *10*(1), 3503–3513.

Sulehu, M., Wisda, W., Wanita, F., & Markani, M. (2025). Optimasi Prediksi Kelulusan Mahasiswa Menggunakan Random Forest untuk Meningkatkan Tingkat Retensi. *Jurnal Minfo Polgan*, *13*(2), 2364–2374. https://doi.org/10.33395/jmp.v13i2.14472

Tribuana, D., Baharuddin, B., & Muhammad Resky, A. (2025). Penerapan Algoritma XGBoost Untuk Prediksi Kepuasan Pelanggan Pada Layanan E-Commerce: Studi Pada Dataset Transaksi Nyata. *Jurnal Teknologi Dan Bisnis Cerdas*, *1*(1), 50–59. https://doi.org/10.64476/jtbc.v1i1.5

Ulumando, M. I. (2025). Model Prediksi Prestasi Mahasiswa ITAF Kupang Menggunakan Backpropagation Berbasis Data Perilaku Belajar. *Jurnal Nasional Komputasi Dan Teknologi Informasi (JNKTI)*, *8*(4), 1770–1781.

Ulumando, M. I. (2026). Klasifikasi Resiko DropOut Mahasiswa ITAF Kupang Menggunakan Random Forest Sebagai Sistem Peringatan Dini. *Simkom (Sistem Informasi Dan Sistem Komputer)*, *11*(1), 116–130. https://doi.org/https://doi.org/10.51717/simkom.v11i1.1255