Journal of Applied Data Sciences
Vol. 7, No. 2, May 2026, pp. 1466-1482

ISSN 2723-6471
1466

Optimizing XGBoost with Optuna for Attendance-Based Prediction of Student Academic Success

Marcello Roy[1], Ririn Ikana Desanti[2,*], Iwan Prasetiawan[3], Suryasari[4]

[1,2,3,4]*Information Systems Study Program, Universitas Multimedia Nusantara, Gading Serpong, Tangerang 15810, Indonesia*

(Received: October 10, 2025; Revised: December 5, 2025; Accepted: March 15, 2026; Available online: May 3, 2026)

Abstract

The growth of data analytics in higher education has increased the need for predictive models that identify students' academic potential and support data-driven decision-making. Academic institutions are able to enhance student outcomes by implementing suitable interventions that are based on an accurate early prediction of Grade Point Average (GPA). This study aims to develop a practical and accurate GPA prediction model based on Extreme Gradient Boosting (XGBoost), enhanced through Optuna-based hyperparameter tuning, to support academic monitoring systems in higher education. This model is intended to assist academic monitoring systems in higher education. Using academic performance data, attendance records, and course load information obtained from 961 undergraduate students, a quantitative predictive modeling approach was implemented. The CRISP-DM framework was implemented during the modelling process, which included the following stages: data understanding, data preparation, modelling, evaluation, and deployment. To ensure the stability and relevance of the model, exploratory analysis and correlation assessments were implemented to determine feature selection. Optuna was employed to optimize hyperparameters, utilizing Bayesian optimization with adaptive trial pruning to efficiently examine the parameter space. Experimental results demonstrate that the Optuna-tuned XGBoost model achieved superior predictive performance compared to baseline XGBoost models and models optimized using Grid Search and Random Search. The proposed model attained a coefficient of determination (R^2) of 0.8637 and a Root Mean Square Error (RMSE) of 0.1165, indicating improved accuracy and robustness in handling large prediction errors. To enhance practical applicability, the final model was deployed in a Streamlit-based web application that enables real-time GPA prediction and supports academic advisors. Overall, the findings confirm that Optuna-based hyperparameter tuning significantly enhances XGBoost performance and provides a solution for data-driven academic monitoring in higher education institutions.

Keywords: Intention Educational Data Mining, Hyperparameter Tuning, GPA prediction, Optuna Optimization, XGBoost

1. Introduction

Higher education plays a strategic role in shaping high-quality human resources with global competitiveness. Student academic achievement is one of the main indicators reflecting the effectiveness of the education system and serves as a reference for academic policy formulation by universities [1], [2]. Various internal factors such as motivation, time management, and learning strategies, as well as external factors such as lecture attendance, socioeconomic conditions, and the learning environment, contribute to academic success [3], [4].

Along with the advancement of information technology, conventional approaches to evaluating student academic performance have transformed into data-driven approaches. The utilization of Big Data Analytics enables educational institutions to efficiently process large volumes of data and gain more accurate insights into the factors that influence student success, particularly in supporting sustainable and effective learning systems [5], [6] ,[7]. Given the multivariable and complex nature of the data, advanced analytical methods such as Machine Learning are required to identify patterns and predict academic performance with precision [8].

In this context, the application of machine learning algorithms has proven relevant, especially in classification and regression cases involving academic data. Several previous studies have demonstrated that algorithms such as Decision Tree, Support Vector Machine, and Random Forest perform well in modeling academic factors [9], [10]. In recent years, the Extreme Gradient Boosting algorithm has emerged as one of the most reliable approaches in educational predictive research due to its high accuracy and superior predictive performance [11]. XGBoost is an optimized implementation of the boosting technique based on the gradient boosting framework. This model works by constructing

***Corresponding author: Ririn Ikana Desanti (ririn.desanti@umn.ac.id)**
DOI: https://doi.org/10.47738/jads.v7i2.1116

an ensemble of decision trees sequentially, where each new model is aimed at correcting the errors of the previous ones [12]. The advantages of XGBoost lie in its use of L1 and L2 regularization to prevent overfitting, as well as its parallel computing capabilities, making it ideal for large and complex datasets such as student academic data.

Optuna has emerged as one of the modern frameworks capable of handling hyperparameter tuning processes efficiently and adaptively. With a define-by-run approach and built-in pruning mechanisms to terminate unpromising trials early, Optuna offers flexible Bayesian-based optimization for various machine learning models, including XGBoost [13], [14]. This makes Optuna the primary choice in this study to evaluate predictive model performance.

Several previous studies have shown the success of the XGBoost and Optuna combination in educational contexts, including learner performance prediction, graduation forecasting, and dropout identification [13], [14]. In the academic domain, this algorithm has also begun to be adopted to predict student learning performance based on parameters such as average grades, attendance, and total credit taken [15]. However, studies that specifically examine the effectiveness of the XGBoost and Optuna combination in the context of predicting students' final GPA in Indonesia remain very limited.

In addition to the relevance of the models used, the methodological approach also plays a crucial role in the success of data-driven research. The CRISP-DM methodology is employed in this study because it provides a systematic phase from business understanding to deployment [16]. In this research, CRISP-DM facilitates the integration of processes from academic data exploration, feature preprocessing, modeling, evaluation, and deployment in the form of a web-based predictive application.

This study develops a GPA prediction model using XGBoost enhanced through Optuna hyperparameter tuning, deployed via a Streamlit-based web application for real-time predictions accessible to university stakeholders such as academic advisors and student affairs departments [17]. The dataset comprises 961 undergraduate students with numerical features including average grades, attendance rates, and course loads. All features were standardized, and cross-validation was implemented to ensure model generalization and prevent overfitting [18]. The study evaluates and compares model performance before and after tuning using regression metrics including Mean Absolute Error, Mean Absolute Percentage Error, Root Mean Squared Error, and R^2 to measure prediction accuracy and precision [19].

The focus of this study is to evaluate and compare the performance of the XGBoost model before and after the tuning process using Optuna. The evaluation is conducted using regression metrics such as Mean Absolute Error, Mean Absolute Percentage Error, Root Mean Squared Error, and R^2 to measure the accuracy and precision of final GPA predictions [19]. Through a quantitative approach and the use of actual data, this study aims to produce a model that is not only statistically accurate but also operationally relevant.

2. Literature Review

The overall workflow of predictive modeling follows the general supervised learning process. Supervised learning has been widely adopted in educational data mining to model and predict student academic performance based on historical data. In this paradigm, labeled academic records—such as grades, attendance, and course-related information—are used to train predictive models that learn the relationship between input features and target outcomes. The general supervised learning workflow typically consists of data collection, preprocessing, model training, evaluation, and prediction, forming the foundation of many predictive analytics studies in higher education. This conceptual workflow, which underpins various machine learning approaches discussed in the literature.

Prediction of student academic success is one of the key topics in the field of educational data mining due to its strategic value in supporting data-driven decision-making in higher education institutions. Various approaches have been developed to build accurate predictive models, ranging from classical algorithms such as Naive Bayes, Decision Tree, and Random Forest to modern approaches based on deep learning and ensemble models.

One of the most widely used approaches in academic prediction studies is the application of the Extreme Gradient Boosting algorithm, which is known for its ability to handle multivariable data, resistance to overfitting, and computational efficiency [9], [14]. Beyond educational applications, XGBoost has also been extended with auxiliary learning components such as autoencoders to enhance predictive performance in complex classification tasks, demonstrating the flexibility and robustness of gradient boosting models across different application domains [13]. However, the performance of XGBoost heavily depends on the selection of model parameters, commonly referred to as hyperparameters. Parameters such as tree depth, learning rate, and L1 and L2 regularization have a significant impact on the final model performance. Without proper tuning, the model is prone to high bias or variance [12].

To optimize parameter selection, recent studies have adopted hyperparameter tuning approaches based on Bayesian optimization, one of which utilizes a framework called Optuna. Unlike static approaches such as grid search and random search, Optuna employs a define-by-run strategy and pruning mechanism to terminate unpromising trials early. This allows Optuna to explore the parameter space efficiently and adaptively [19], [20]. A study demonstrated that the use of Optuna significantly enhances the performance of boosting algorithms such as XGBoost, LightGBM, and CatBoost in the context of dropout prediction and student academic success, particularly when dealing with imbalanced data.

Several other studies have attempted to improve the accuracy of predictive models by combining multiple algorithms through ensemble or hybrid approaches, including hybrid deep neural network architectures designed for student performance prediction [21]. In addition to algorithmic modelling, behavioural and technology adoption perspectives have also been explored in educational analytics. Lestari et al. examined students' digital entrepreneurial behavior using an integrated TPB-UTAUT framework, demonstrating how data-driven analysis can be used to model student engagement and behavioral factors in higher education context [22]. Other studies have applied deep learning architectures, such as the combination of BiLSTM with attention mechanisms, to predict student's final grades, achieving accuracy level above 90% [23]. However, this kind of approach tends to be more complex, require longer training times, and is not always feasible for direct implementation within web-based academic information systems.

Compared to ensemble-based or deep hybrid network approaches, the use of Optuna in XGBoost offers advantages in terms of efficiency, interpretability, and easier integration into real-world applications. In the present study, the XGBoost model tuned with Optuna achieved a coefficient determination score of 0.8637 and an RMSE of 0.1165, outperforming baseline configurations and traditional tuning methods such as Grid Search and Random Search. The tuning process involved optimizing parameters such as max_depth, learning_rate, lambda, and alpha using a Bayesian optimization approach. The final model was then implemented in a Streamlit-based web application, enabling GPA predictions and supporting faster decision-making by academics. Therefore, this study is aimed at systematically exploring how Optuna-based hyperparameter tuning can significantly and efficiently improve the performance of GPA prediction models, as will be further elaborated in the following methodology section.

3. Methodology

The methodology adopted in this study follows the CRISP-DM framework, which provides a comprehensive, industry-standard approach for developing data mining projects and has been widely applied in educational performance prediction studies [16], [24]. As shown in figure 1, the workflow is structured into six interrelated phases: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment. Each phase is implemented iteratively to ensure model robustness, relevance to the academic domain, and readiness for real-world deployment. Several CRISP-DM phases were iteratively revisited, especially during data preprocessing, outlier handling, and hyperparameter tuning, until stable and satisfactory model performance was achieved.

The process illustrated in figure 1 represents an automated learning pipeline that integrates XGBoost with Optuna to find the most accurate configuration for predicting student success. It shifts away from manual trial-and-error by using an intelligent feedback loop that balances exploration of new settings with the exploitation of known successful parameters.

The cycle begins with Optuna acting as the "brain" of the operation. Instead of trying every possible combination, it uses a sampler to intelligently select values for hyperparameters like learning rate, max depth, and regularization terms. These selections are passed into the XGBoost algorithm, which serves as the core predictive engine. This stage is crucial because it defines the search space for the student performance predictors such as average grade, attendance, and course load.

Once the parameters are set, the XGBoost model undergoes training and validation. A key feature shown in architecture is the pruning callback. As the model trains, Optuna monitors its intermediate performance. If a specific set of parameters is performing significantly worse than previous versions, Optune "prunes" or terminate that trial early. This prevents the system from wasting computational resources on unpromising models, making the training phase much faster than traditional grid search methods.

After the training is completed, the model's performance measured by the objective value is fed back into the Optuna study database. This creates a history of trials. Optuna analyzes this historical data to refine its next set of predictions, narrowing down the search until it identifies the best model. This final, optimized model is then ready for deployment, providing the high-precision GPA predictions used in the web application.

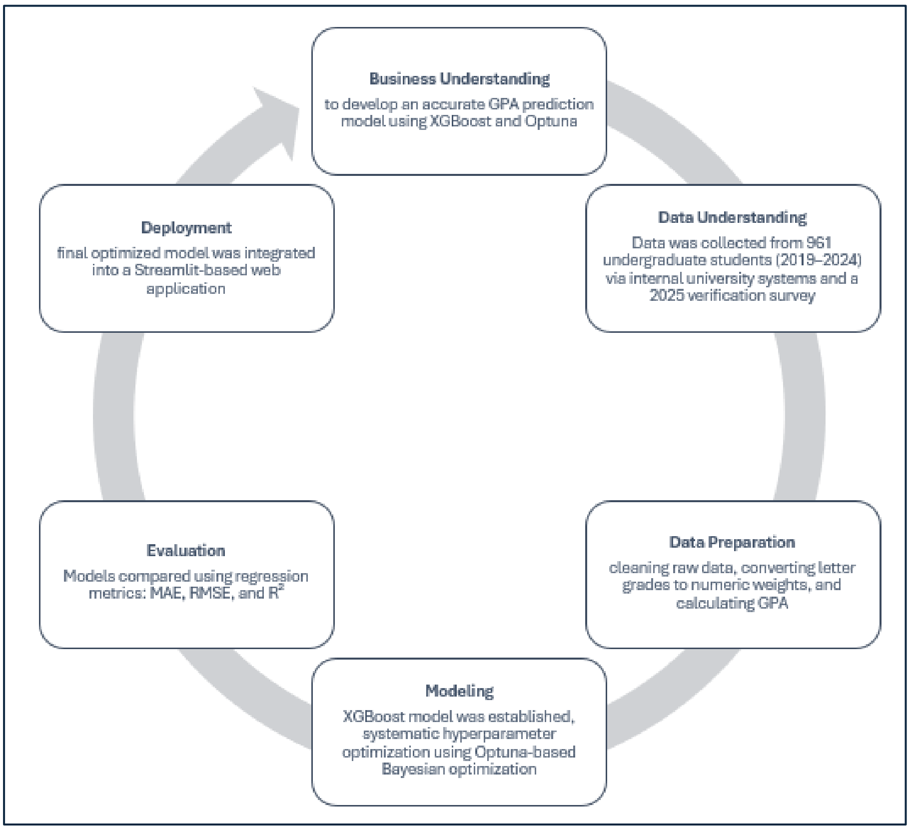

Figure 1. CRISP-DM Framework

The initial phase of this research involves a comprehensive understanding of the problem context. In recent years, higher education institutions have increasingly adopted data-driven approaches to support decision-making and optimize learning outcomes [20], [25]. Within this context, predictive analytics has emerged as an effective decision-support mechanism to enhance operational performance and enable personalized recommendations, highlighting the practical value of data-driven models for real-time decision-making systems across various organizational settings. The proposed model architecture integrates the Extreme Gradient Boosting (XGBoost) algorithm with Optuna, an open-source hyperparameter optimization framework, to achieve high-precision GPA predictions.

3.1. Integration of Optuna and XGBoost

The core of the architecture lies in the Bayesian optimization approach provided by Optuna, which replaces traditional trial-and-error tuning. Define-by-Run Strategy: Unlike static methods, Optuna utilizes a define-by-run approach, allowing the search space for hyperparameters to be constructed dynamically during runtime. Hyperparameter Search Space: The optimization process focuses on eight key XGBoost parameters such as the number of estimators, maximum tree depth, learning rate, subsample, colsample_bytree, gamma, reg_alpha (L1 regularization), and reg_lambda (L2 regularization). Objective Function: Optuma aims to maximize the coefficients of determination (R^2) by iteratively sampling parameter combinations that are predicted to yield better performance based on previous trial results. Adaptive Trial Pruning: To ensure computational efficiency, the architecture includes an integrated pruning mechanism. This automatically terminates "unpromising" trials (parameter sets performing poorly in early stages) to focus resources on more successful configurations.

3.2. Comparison Models and Approaches

To rigorously evaluate the effectiveness of the proposed Optuna-XGBoost model, it was compared against three distinct approaches using identical train-test splits and feature normalization: Baseline XGBoost, A reference model utilizing default parameters to establish a performance benchmark without tuning. Grid Search Optimization, A

ISSN 2723-6471

conventional approach that relies on an exhaustive search across 12 manually predefined parameter combinations, offering limited coverage of the search space. Random Search Optimization, A stochastic approach that randomly samples the parameter space (30 trials) to provide a more flexible but less structured comparison to the Bayesian method.

By comparing those methods, the study demonstrates that while Grid Search may minimize average absolute error (MAE), the Optuna-tuned model provides superior robustness by significantly reducing the Root Mean Square Error (RMSE), which is critical for identifying at-risk students with large prediction deviations.

3.3. Data Collection and Sampling Strategy

Data were collected from two primary sources: historical academic records from the university's internal system (2019–2024) and supplementary biodata verification via a Google Form survey conducted in May 2025. To ensure the model's robustness and its ability to adapt to changing academic environments, this study leverages data characterized by significant temporal diversity. The primary dataset comprises extensive historical records from 2019-204, providing the necessary volume and longitudinal pattern to train the XGBoost engine on long term student success factors. This historical foundation is supplemented and validated by primary data collection effort conducted via a verification survey in May 2025. By integrating five years of retrospective institutional data with real-time academic biodata from 2025, the methodology minimizes data drift which is the phenomenon of predictive accuracy degrades over time, thereby ensuring that the resulting model remains relevant for contemporary academic monitoring and future student cohorts.

This study employed purposive sampling to select participants meeting specific criteria essential for predicting student academic success. This non-probability sampling approach targets information-rich cases to improve reliability and credibility [26]. The sample comprised 961 undergraduate students from the Information Systems program at the Faculty of Engineering and Informatics, spanning academic years 2019–2024. Only students enrolled in semesters 1–8 with complete academic records and active enrollment status were included, while transfer students (due to incomplete historical records), those on academic leave, and records exceeding 30% missing data were excluded. To mitigate potential sampling bias, the selection incorporated multiple cohorts across the specified years for temporal diversity and represented all semester levels from 1 to 8, thereby capturing various stages of academic progression. The focus on a single program helped control differences in curriculum structure, course requirements, and evaluation standards, strengthening internal validity while preserving methodological applicability to comparable programs. The resulting dataset consisted of 961 students with fully complete academic records.

Table 1 summarizes the data sources used in this study, which consist of academic records collected from the institutional academic information system. The dataset includes student performance data such as grades, attendance records, course credit information, and enrolment history, which together provide a comprehensive representation of student's academic activities. These data sources were selected to ensure that both academic achievement and learning engagement factors were adequately captured for subsequent analysis and modelling.

Table 1. Data Understanding

Data Period	Rank	Source	Data Name	Description
2019–2024	1	UMN Internal System	UMN Student Academic Records	Historical academic data of active students from internal systems
2025	2	Google Form Survey (May 10)	Academic Biodata Verification	Supplemental survey to update and validate academic profiles

The combination of these two datasets provides a comprehensive foundation for analysis. Historical records from 2019 to 2024 offer a rich view of students' academic performance over multiple semesters, while the 2025 survey data ensures recent academic and demographic attributes are up to date. Together, they enable the construction of a well-rounded dataset suitable for GPA prediction using machine learning techniques.

Table 2 provides a comprehensive overview of the dataset structure, encompassing the variables, their description, and data types. This table describes the organization of raw academic data before preprocessing, acting as a reference for understanding the role of each attribute in the analysis. The variables outlined in the table serve as the basis for subsequent data cleaning, transformation, and feature selection processes undertaken during the data preparation phase.

Table 2. Data frame List

Column Name	English Term	Definition
NIM	student_id	Student Unique Identifier.
angkatan	enrolment_year	The year the student was enrolled.
semester	active_semester	The active semester at the time.
sks	credits	Number of credits for the course.
nilai	grade	The final score achieved in a specific course.
Mata kuliah	subject	The name of the course.
IPK	GPA	Cumulative Grade Point Average.
rata2_nilai	avg_grade	The average of the student's grades across courses.
rata2_hadir	avg_attendance	The average attendance rate of the student.
Jumlah kehadiran	total_attedance	Total attendance.
jumlah mata kuliah diambil	course_taken	Total number of attended class sessions.

Most columns contain final grades and attendance records for each course, following naming patterns such as "Accounting Principles 1 (grade)" and "Accounting Principles 1 (attendance)". Demographic attributes include date_of_birth, gender, region_of_origin, father's_education, and mother's_education. For analytical purposes, these were restructured into consolidated variables such as average grade, average attendance, and total courses taken. Based on the variables outlined in Table 2, a sequence of preprocessing steps was applied to ensure data quality and model readiness.

Table 3 illustrates the overall structure of the academic data set used in this study. Each row represents an individual student, while columns summarize aggregated academic attributes derived from course-level grades and attendance records. Historical academic data from 2019-2024 were combined with survey-based verification in 2025, then transformed into consolidated features such as average grade, average attendance, and number of courses taken. This structure ensures data completeness and suitability for predictive modelling.

By understanding the structure and quality of the data, this step serves as a crucial foundation for the subsequent data preprocessing phase, which aims to optimally prepare the data before training the model using Machine Learning algorithms.

Table 3. Dataset Structures

NIM	Cohort	Semester	Accounting Principles 1 (grade)	Accounting Principles 2 (attendance)	Introduction to Business Management (grade)	Introduction to Business Management (attendance)	Programming Fundamentals (grade)	…	Marketing Research (attendance)	Film Distribution & Exhibition (grade)
10	2019	1911	55.0	13.0	68.0	14.0	72.0	…	NaN	NaN
12	2019	1911	NaN	NaN	80.0	14.0	92.0	…	NaN	NaN
35	2019	1911	65.0	14.0	75.0	14.0	76.0	…	NaN	NaN
83	2019	1911	78.0	14.0	82.0	14.0	93.0	…	NaN	NaN
107	2019	1911	NaN	NaN	70.0	14.0	72.0	…	NaN	NaN

The target variable, Grade Point Average (GPA), is measured on a continuous scale from 0.00 to 4.00. To enhance interpretability and practical utility for stakeholders such as academic advisors, GPA predictions are mapped to official university graduation predicates. These classifications follow the institutional standards outlined in the university's Academic Guidance for Undergraduate Programs.

Table 4. GPA Classification Standards

Graduation Predicate	GPA Range
With Distinction	3.51 – 4.00
Very Satisfactory	3.01 – 3.50
Satisfactory	2.76 – 3.00
Below Standard	< 2.76

Based on table 4, those thresholds provide a meaningful context for model outputs, transforming numerical GPA estimates into actionable categories aligned with official achievement levels. For example, a predicted GPA of 3.60 would be classified as "Cum Laude," signaling exceptional performance and potential eligibility for academic honors. The classification scheme was validated against historical distributions in the dataset, ensuring alignment with observed student outcomes.

This mapping supports early intervention strategies, as predictions falling into "Needs Attention" can prompt timely academic support, like how confidence intervals and statistical significance guide decision-making in experimental evaluations.

3.4. Data Preparation and Modeling

The data preparation stage serves as a critical bridge between data understanding and model training, focusing on transforming raw datasets into a structured format. To handle the academic dataset efficiently, an automated pattern-matching technique was implemented to categorize features. The system first identifies all columns representing course grades by filtering for the suffix '(nilai)', storing these in a specific grade list.

Using these identified grade headers as a baseline, the process programmatically generates a corresponding list for attendance by replacing the grade suffix with '(hadir)', ensuring perfect alignment between a student’s performance and their presence in a specific course. Finally, the core course names are extracted by stripping the suffixes entirely. This automated mapping ensures that subsequent preprocessing steps—such as converting letter grades to numeric weights and calculating final GPAs—are performed accurately across all subjects without manual column mapping.

The first step involved identifying and separating the column names related to grades and attendance for each course. Columns containing the string "(nilai)" were identified as grade features, and corresponding attendance columns were automatically generated by replacing the "(nilai)" label with "(hadir)". From these column pairs, the course names were extracted for the purpose of aggregating grades into a GPA-compatible format.

As shown in table 5, all identified attendance columns were systematically validated and converted to numeric format to ensure computational compatibility. This step aligns attendance records with credit unit (SKS) information according to institutional curriculum standards prior to GPA computation. Non-numeric or invalid entries were automatically treated as missing values to maintain data integrity. A total of 209 attendance columns were successfully standardized and validated, enabling reliable aggregation, minimum attendance verification, and seamless integration into the subsequent predictive modeling pipeline.

Table 5. Added SKS (Credits) Column

No.	Course Name	Course Credits (SKS)
1	Introduction to Business Management	3
2	Programming Fundamental	3
3	Management Information System	3
4	Introduction to System and Information Technology	3

Building upon the attendance validation step, table 6 confirms that all grade and attendance attributes have been successfully standardized into numerical formats. All attendance and grade features were converted into consistent numeric formats, with invalid entries handled as missing values, ensuring data integrity and compatibility with subsequent preprocessing and modeling steps.

Table 6. Check Datatypes After Mapping

Attribute	Data Type
Accounting Principles 1 (attendance)	Float64
Introduction to Business Management (attendance)	Float64
Programming Fundamentals (attendance)	Float64
Introduction to System and Information Technology (attendance)	Float64
Commercial Production (attendance)	Float64
News Graphics and Design (attendance)	Float64
Marketing Research (attendance)	Float64
Film Distribution & Exhibition (attendance)	Float64
Industrial Automations (attendance)	Float64

The next step in the data preparation process is converting numerical course scores into letter grades, which will be used for GPA calculation. This process is carried out by defining a bobot_ipk dictionary, which serves as a reference for mapping letter grades to their corresponding GPA weight values.

After the grade conversion function is defined, the next step is to apply it across all grade and attendance columns for each course. This process involves restructuring the dataset by iterating through pairs of course names, grade columns, and attendance columns, then inserting a new column containing the corresponding letter grades. Additionally, the system adds a column for the number of credit hours by matching course names with the data in the sks_df DataFrame.

The next step is to calculate the Grade Point Average for each student. This calculation is performed by summing the total weighted scores of all courses based on the letter grades and their corresponding credit hours. Each letter grade is converted into a GPA weight using the bobot_ipk dictionary, then multiplied by the number of SKS. The total weighted score is then divided by the total valid SKS to obtain the final GPA value as shown below in table 7.

Table 7. Data Calculation for GPA

No	GPA	Grades Average	Attendance Average	Number of Courses Taken
0	3.455128	78.000000	11.645833	48
1	3.811441	85.541667	12.333333	48
2	3.714286	81.893617	12.617021	47
3	3.868644	86.020408	12.551020	49
4	3.565574	80.354167	12.500000	48

The GPA column is calculated by multiplying the letter grade weights by the credit hours for each course, then dividing the result by the total valid units. Meanwhile, average_grade and average_attendance are computed as the mean values across all courses taken by each student. The feature total_courses_taken indicates the number of courses with non-missing grade values. These three features will serve as predictors in the academic modeling process in the subsequent phase. To improve data quality before modeling, outlier detection and removal were performed on the numerical features. This step aims to prevent model distortion caused by extreme values that may skew data distribution. The numerical features analyzed include average_grade, average_attendance, total_courses_taken, and GPA. The GPA is calculated as a weighted average of course grades based on credit units, as defined in Equation (1):

$$GPA = \frac{\sum_{i=1}^{n}(G_i x\ C_i)}{\sum_{i=1}^{n} C_i} \quad (1)$$

G_i represents the numeric grade obtained in course *i*, and C_i denotes the corresponding course credit (SKS), and *n* is the total number of courses taken by a student.

Next step is to identify the most predictive variables while minimizing multicollinearity, Pearson correlation analysis was conducted on the derived features and the target variable (GPA). This analysis serves two purposes: (1) identifying

ISSN 2723-6471

features with strong predictive power, and (2) detecting redundant features that would not contribute additional information to the model.

Figure 2 demonstrates the bivariate relationships between each candidate feature and GPA through scatter plots with fitted linear trend lines. The left panel shows a strong positive linear relationship between Average Grade and GPA (r = 0.889), with data points closely following the trend line, indicating that students with higher course grades consistently achieve higher GPAs. The middle panel displays a moderate positive correlation between Average Attendance and GPA (r = 0.307), showing that while attendance contributes to academic success, the relationship is less deterministic than grades. The right panel reveals a weak but positive correlation between Courses Taken and GPA (r = 0.203), suggesting that academic workload has a limited direct impact on GPA within the observed range.

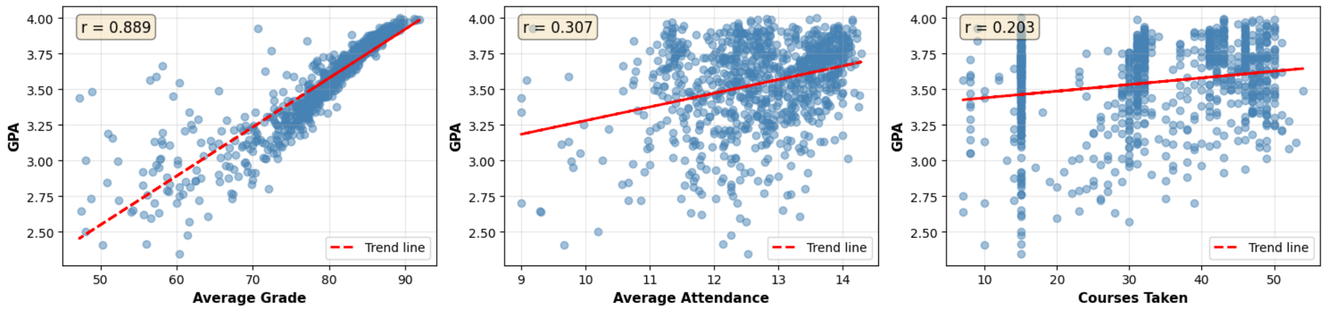

Figure 2. Relationship between Features and GPA

Figure 3 presents a comprehensive correlation matrix examining relationships among all numerical features and the target variable. The heatmap uses color intensity to indicate correlation strength, with dark red representing strong positive correlation, white indicating no correlation, and blue indicating negative correlation. The matrix reveals that Average Grade demonstrates the strongest correlation with GPA (r = 0.89), followed by Average Attendance (r = 0.31) and Courses Taken (r = 0.20). Importantly, the inter-feature correlations are relatively low (ranging from -0.33 to 0.43), indicating minimal multicollinearity among the selected predictors. Based on this analysis, average grade, average attendance, and number of courses taken were selected as final predictors due to their strong association with GPA and minimal multicollinearity.

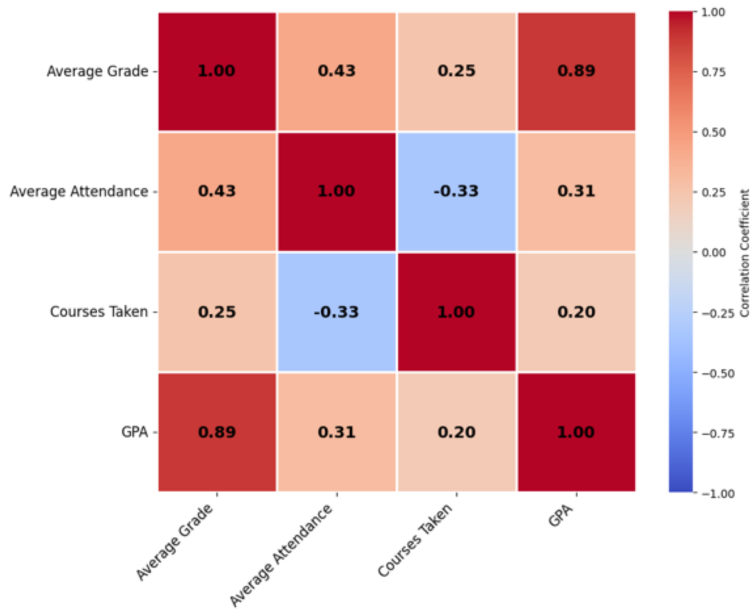

Figure 3. Correlation Matrix of Numerical Features

To improve data quality before modeling, outlier detection and removal were performed on the numerical features, including Average Grade, Average Attendance, Courses Taken, and GPA. This study employed the Z-score method to identify and handle extreme values. A threshold of $|Z| > 3$ was applied, where data points falling outside three standard deviations from the mean were classified as outliers and subsequently removed. This conservative threshold ensures

ISSN 2723-6471

that only extreme anomalies are eliminated while preserving the natural variance of the student dataset. As shown in figure 4, this process successfully mitigated potential model distortion without compromising the overall data distribution. It shows distributions before and after removal. Box plots demonstrate elimination of extreme values while preserving overall distribution characteristics with minimal impact on median and interquartile range.

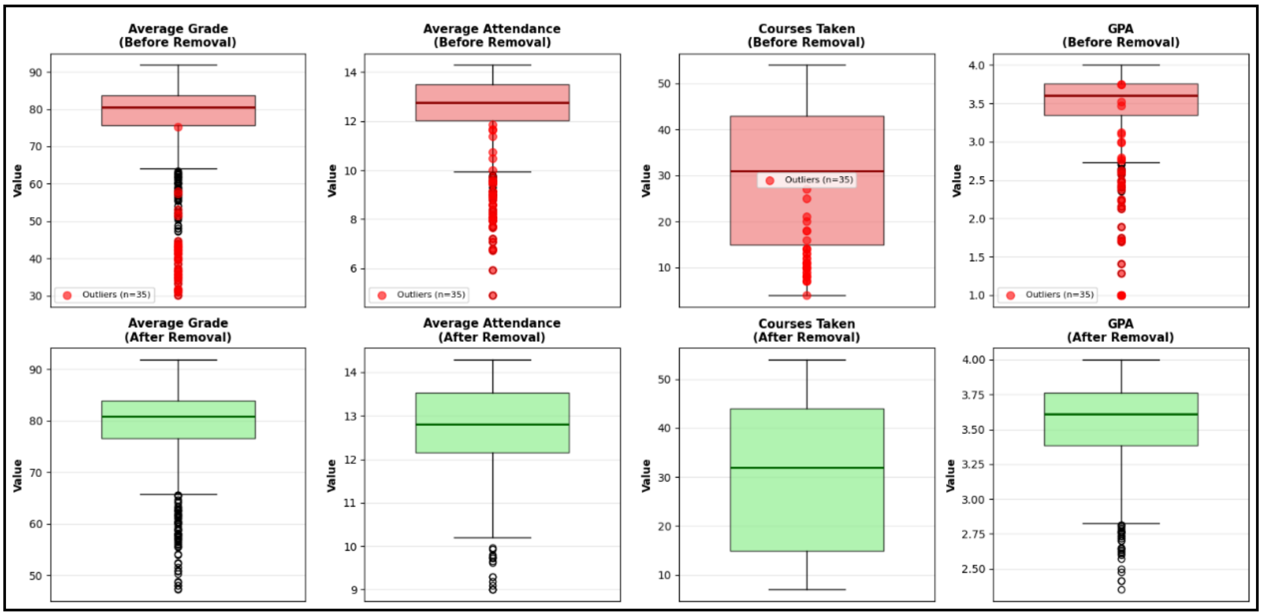

Figure 4. Data Before and After Outlier Removal

After the data cleaning process is completed, the next step is to split the dataset and build a preprocessing pipeline before modeling. In this phase, the features used as predictors are average_grade, average_attendance, and total_courses_taken, while the target variable to be predicted is the GPA. The data is divided into two parts: 80% for training and 20% for testing, using the train_test_split function with stratification by GPA quartiles to maintain representative distributions in both sets, resulting in 768 training samples and 193 test samples. The dataset split into training (80%) and testing (20%) subsets using stratification based on GPA quartiles. This approach ensures balanced representation of academic performance levels and supports reliable model evaluation.

To ensure that all numerical features are on a consistent scale, a standardization process was applied using StandardScaler() through a Column Transformer. This entire process was organized into a preprocessing pipeline to maintain efficiency and structure before feeding the data into the XGBoost model, which will be trained and evaluated in the subsequent modeling stage.

The modeling phase was conducted to develop a predictive model for estimating students' Grade Point Average based on previously prepared numerical features, such as average grade, average attendance, and the number of courses taken. Initially, the baseline XGBoost model was trained to obtain a preliminary performance benchmark. The model was trained on the training dataset and evaluated on the test dataset. The XGBoost model minimizes a regularized objective function, defined as follows:

$$\mathcal{L}(\emptyset) = \sum_i l(\hat{y}_i, y) + \sum_k \Omega(f_k) \quad (2)$$

where l is a differentiable convex loss function that measures the difference between the prediction $\hat{y}_i$ and the target y_i , and Ω represents the regularization term that penalizes the complexity of the model to prevent overfitting.

Subsequently, hyperparameter optimization was performed using Optuna, an open-source library that enables efficient search for the best parameter combinations through a Bayesian optimization approach [27]. The tuned parameters included the number of estimators, maximum tree depth, learning rate, subsample, colsample_bytree, gamma, reg_alpha, and reg_lambda. The search was conducted over 30 trials with the objective of maximizing the Coefficient of Determination (R^2), calculated as:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \quad (3)$$

This metric represents the proportion of variance in the student's GPA that is predictable from the independent variables. This tuning process allowed efficient exploration of the parameter space without the need for exhaustive search. Furthermore, Optuna's integrated pruning mechanism was applied to automatically terminate unpromising trials at an early phase, thereby improving optimization efficiency.

4. Results and Discussion

This study evaluates and compares multiple hyperparameter optimization approaches for predicting student GPA using the XGBoost algorithm. The comparison covers the baseline model with default parameters, Grid Search, Random Search, and Optuna-based Bayesian optimization. All models are implemented within a standardized preprocessing pipeline using StandardScaler for feature normalization and evaluated on identical train-test splits to ensure fair and reproducible results. All tuning methods (except baseline) are constrained to comparable computational budgets where possible, using 5-fold cross-validation with R^2 as the primary objective. The best configuration from each method is retrained on the full training set and evaluated on the held-out test set.

4.1. XGBoost – Baseline Model

A baseline model was established using XGBoost Regressor with default parameters to assess initial predictive capability without tuning overhead. This baseline serves as a reference point for evaluating the effectiveness of subsequent hyperparameter optimization efforts.

Table 8 illustrates the baseline XGBoost modeling pipeline with standardized input features and default hyperparameters. This model serves as a reference point for evaluating the impact of subsequent hyperparameter optimization methods.

Table 8. Model XGBoost - Baseline

Metric	**Value**
R^2	0.8086807727813721
MAE	0.07926927506923676
MAPE	2.4321129504398895
RMSE	0.13800799262096874

The baseline model pipeline integrates StandardScaler preprocessing with the XGBoost regressor, ensuring consistent feature scaling before model training. The pipeline architecture enables proper cross-validation implementation where preprocessing is fitted on training folds only, preventing data leakage during evaluation.

Based on the table 9, the baseline XGBoost model demonstrated robust predictive power, explaining 83.12% of GPA variance $R^2 = 0.8312$ with minimal computational overhead 0.05 seconds. The Mean Absolute Error of 0.0753 indicates that predictions deviate from actual GPA values by approximately 0.075 points on average, equivalent to 2.30% relative error on a 4.0 GPA scale. The Root Mean Squared Error of 0.1296 suggests that while most predictions are accurate, occasional larger errors exist, as RMSE penalizes extreme deviations more heavily than MAE due to squaring. However, the gap between MAE 0.0753 and RMSE 0.1296 indicates prediction variance, with some students experiencing larger prediction errors than others. Additionally, the R^2 value of 0.8312, while strong, leaves approximately 17% of GPA variance unexplained, suggesting potential for improvement through hyperparameter optimization. These observations motivated systematic exploration of Grid Search, Random Search, and Optuna tuning approaches.

Table 9. Model XGBoost - Baseline

Model	**MAE**	**MAPE**	**RMSE**	**R^2**	**Time**
XGBoost (Baseline)	0.0753	2.30%	0.1296	0.8312	0.05s

4.2. XGBoost with Grid Tuning

Grid Search was employed to systematically explore predefined parameter space through exhaustive evaluation of all parameter combinations. This method provides comprehensive coverage of specified parameter ranges, ensuring no combination within the grid is overlooked.

Based on the evaluation results shown in table 10, the XGBoost model tuned using Grid Search achieved an R^2 score of 0.8383, indicating that approximately 83.8% of the variance in student GPA can be explained by the model. The MAE, MAPE, and RMSE values are relatively low, suggesting that the tuned model is able to generate accurate predictions and capture the relationship between input features and the target variable effectively.

Table 10. Model XGBoost – Grid Search Tuning

Metric	Value
R^2	0.8383736610412598
MAE	0.0764264315366745
MAPE	2.326851202642795
RMSE	0.12684722298000961

The Grid Search–optimized model recorded an MAE of 0.0764 and an RMSE of 0.1268, reflecting a low average prediction error and a limited presence of large deviations. These results indicate that Grid Search is effective in improving model stability and predictive precision. However, compared to more adaptive optimization methods, its performance gains are constrained by the predefined and discrete nature of the search space, which may limit its ability to identify more optimal hyperparameter configurations. Table 11 summarizes the result of XGBoost Grid Search Tuning.

Table 11. Model XGBoost – Grid Search Tuning

Model	MAE	MAPE	RMSE	R^2	Time
XGBoost + Grid Search	0.0764	2.32%	0.1268	0.8383	12.87s

4.3. XGBoost with Random Search Tuning

Random Search was applied to explore the hyperparameter space by randomly sampling a fixed number of parameter combinations. Unlike Grid Search, this approach does not evaluate all possible combinations but instead provides broader coverage of the search space within a predefined computational budget, making it more efficient for higher-dimensional hyperparameter tuning.

Table 12. Model XGBoost – Random Search Tuning

Metric	Value
R^2	0.8067784905433655
MAE	0.07987713068723679
MAPE	2.4312880114796496
RMSE	0.1386923976272372

Based on the evaluation results shown in table 12, the XGBoost model optimized using Random Search (30 trials) achieved an R^2 score of 0.8068, indicating that approximately 80.6% of the variance in student GPA can be explained by the model. The MAE, MAPE, and RMSE values remain relatively low, suggesting that Random Search can improve predictive performance compared to the baseline model, although the overall accuracy is lower than that achieved by more structured optimization approaches. Table 13 summarizes the result of XGBoost Random Search Tuning. This results further motivate the adoption of adaptive optimization strategies, which are explored in the subsequent Optuna-based tuning.

Table 13. Model XGBoost – Random Search Tuning

Model	MAE	MAPE	RMSE	R^2	Time

ISSN 2723-6471

XGBoost + Random Search	0.0798	2.43%	0.1386	0.8067	20.65s

4.4. XGBoost with Optuna

An Optuna-tuned XGBoost model was developed to enhance predictive performance through adaptive hyperparameter optimization. Optuna applies a Bayesian optimization strategy that iteratively guides the search toward promising regions of the hyperparameter space based on previous trial results, allowing efficient exploration without exhaustive enumeration. The Optuna-based tuning process was integrated within the same preprocessing pipeline as the baseline model, combining StandardScaler with the XGBoost regressor. This ensures consistent feature scaling and fair evaluation, where preprocessing steps are fitted exclusively on training folds during cross-validation to prevent data leakage. Table 14 summarizes the result of XGBoost with Optuna Tuning.

Table 14. Model XGBoost – Optuna Tuning

Model	MAE	MAPE	RMSE	R^2	Time
XGBoost + Optuna	0.0809	2.42%	0.1165	0.8636	10.25s

The Optuna-tuned XGBoost model demonstrated strong predictive capability, explaining 86.37% of the variance in student GPA R^2 of 0.8637. The Mean Absolute Error of 0.0810 indicates that, on average, predictions deviate from actual GPA values by approximately 0.08 points, corresponding to a 2.42% relative error on a 4.0 GPA scale. The Root Mean Squared Error of 0.1165 reflects a lower magnitude of large prediction errors, as RMSE penalizes substantial deviations more heavily than MAE. The relatively smaller gap between MAE and RMSE suggests a more consistent error distribution across students, with fewer extreme prediction errors. Although the optimization process required higher computational time (10.25 seconds) compared to the baseline model, the resulting improvement in explained variance indicates that adaptive hyperparameter tuning effectively enhances the model's ability to capture complex relationships between input features and academic performance outcomes.

4.5. Comparison

The four methods exhibit distinct performance characteristics and trade-offs. The baseline provides a strong, computationally negligible foundation (R^2 = 0.8312) but leaves potential for better generalization untapped. Grid Search delivers modest, reliable gains with the lowest MAE (0.076), favoring conservative regularization. Random Search proves inefficient, underperforming the baseline (R^2 = 0.807) due to overfitting-prone configurations despite higher resource use. Optuna clearly outperforms in key regression metrics (highest R^2 = 0.864, lowest RMSE = 0.117) while remaining efficient through intelligent sampling and early pruning. The performance comparison across all models is visualized in figure 5.

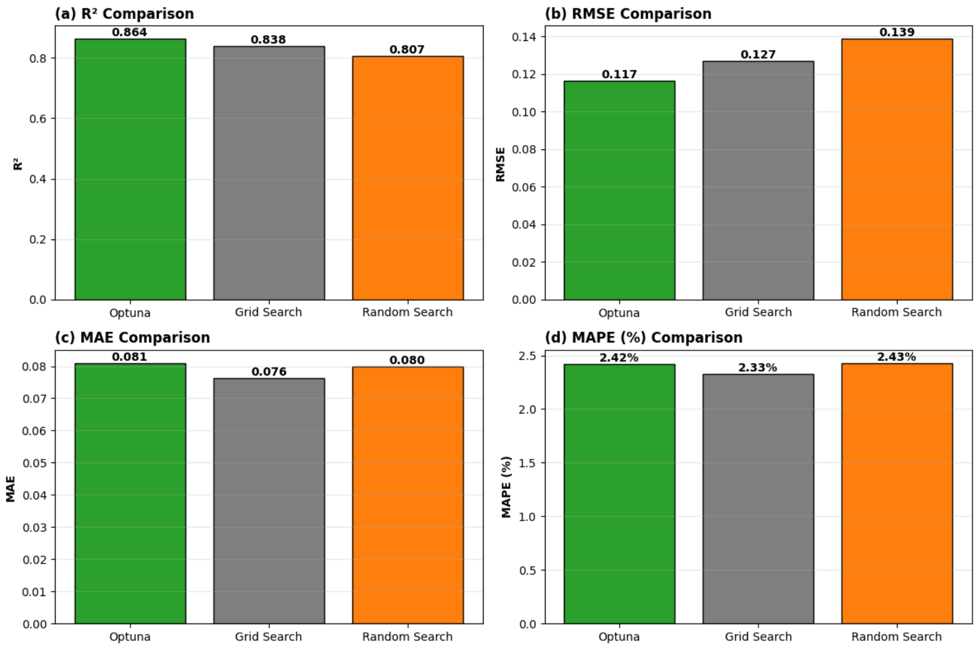

Figure 5. Comparison of Machine Learning Metrics

A critical insight is the trade-off between MAE and RMSE: Grid Search minimizes average absolute error, but Optuna's superior RMSE reduction indicates fewer large prediction errors that are essential for academic early warning systems where extreme mispredictions could lead to inappropriate interventions and erode trust. The overall performance summary is followed by table 15 for a concise side-by-side view including improvements relative to baseline.

Table 15. Comparison of Machine Learning Metrics

Model	Trial	MAE	MAPE	RMSE	R^2	Time
Baseline	-	0.0753	2.30%	0.1296	0.8312	0.05
Grid Search	12 Combinations	0.0764	2.33%	0.1268	0.8384	12.88
Random Search	30	0.0799	2.43%	0.1387	0.8068	20.66
Optuna	30	0.0810	2.42%	0.1165	0.8637	10.25

The baseline model provided a robust foundation, but hyperparameter tuning produced varied outcomes across optimization methods. The baseline XGBoost model achieved a coefficient of determination R^2 of 0.8312, a mean absolute error of 0.0753, and a root mean squared error of 0.1296. Grid Search slightly improved predictive accuracy, achieving an R^2 of 0.8384. While its MAE 0.0764 was slightly higher than the baseline, its RMSE of 0.1268 indicates a better performance in reducing larger prediction deviations.

Random Search resulted in lower overall performance, with an R^2 of 0.8068, an MAE of 0.0799, and an RMSE of 0.1387. These results indicate that while Random Search can explore a broader hyperparameter space, its stochastic nature can lead to inefficient allocation of computational resources and suboptimal configurations in this specific dataset.

Optuna demonstrated the strongest predictive performance among all evaluated approaches. The Optuna-tuned XGBoost model achieved the highest R^2 of 0.8637, along with an MAE of 0.0810 and the lowest RMSE of 0.1165. The significantly reduced RMSE is a key finding, as it indicates fewer extreme prediction deviations, resulting in more consistent accuracy across the student population.

The differentiation between MAE and RMSE is very significant for educational early warning systems. Although the baseline had the lowest average absolute error, the Optuna-tuned XGBoost model achieved a substantial reduction in RMSE compared to both the baseline and traditionally tuned model, indicating improved robustness in handling large prediction errors. Statistical validation using bootstrap confidence intervals at the 95 percent level confirmed the robustness of Optuna's R^2, which remained differentiated from other techniques. Paired statistical testing demonstrated a statistically significant improvement over both the baseline and Grid Search.

Based on these results, the Optuna-optimized XGBoost model was selected to support reliable GPA forecasting and early identification of at-risk students. Future work may investigate ensemble-based approaches or the integration of additional academic and behavioral features to further enhance predictive performance.

4.6. Research Limitations

Despite the superior performance of the Optuna-XGBoost model, several limitations must be acknowledged. First, the dataset is limited to undergraduate students from a single Information Systems program, which may affect the model's generalizability to other academic disciplines with different grading structures. Second, the features used are purely numerical (grades, attendance, and credits); the inclusion of non-academic factors such as psychological well-being, socioeconomic status, or student engagement in extracurricular activities could potentially enhance the model's predictive power. Lastly, the current model relies on historical attendance patterns, which may fluctuate due to changes in university policies or hybrid learning environments. Future research should explore longitudinal data to capture shifting academic trends over time.

4.7. Deployment

To ensure the practical use of the predictive model, a web application was developed using the Streamlit framework. As illustrated in figure 6, the application interface allows users such as academic staff or student advisors to input three key features: average grade, average attendance, and the number of courses taken.

Once the inputs are submitted, the application utilizes the Optuna-tuned XGBoost model to predict the student's final GPA. The prediction result is presented not only as a numerical value but also accompanied by a categorical academic

 ISSN 2723-6471

success classification, making the output more interpretable and actionable. These thresholds were determined based on a historical analysis of academic data patterns, enabling early intervention when necessary. The application further enhances usability by displaying intuitive visuals and textual interpretations of the prediction results, helping users understand how academic consistency (grades and attendance) contributes to predicted performance. A preliminary usability assessment was conducted with academic advisors as potential end users, who reported that the application was easy to use and helpful for early academic monitoring. This feedback indicates the practical applicability of the system in real academic setting.

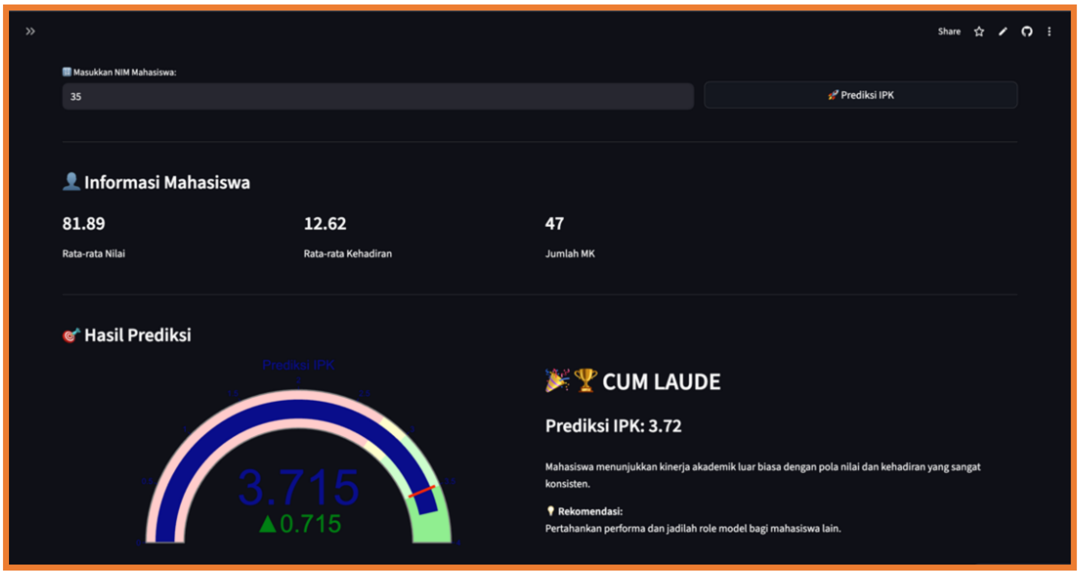

Figure 6. Web Application Deployment Interface

The completion of this deployment stage marks the end of the machine learning lifecycle from data preprocessing, modeling, evaluation, to real-world application integration demonstrating how predictive analytics can be effectively translated into usable academic tools.

5. Conclusion

The evaluation results confirm that hyperparameter tuning using Optuna significantly improved the predictive performance of the XGBoost model. Compared to the baseline model, which yielded an R^2 of 0.8312 and RMSE of 0.1296, the Optuna-tuned model achieved a higher R^2 score of 0.8637 and a lower RMSE of 0.1165. This improvement reflected in a substantial reduction in RMSE, indicates enhanced prediction accuracy and reliability, demonstrating that the proposed model is more robust in handling large prediction errors.

Furthermore, the tuned model demonstrated strong generalization, with consistent performance across regression metrics. By utilizing only key numerical features of average grade, attendance, and number of courses, the model successfully captures core academic behavior patterns. The study provides empirical evidence that the combination of XGBoost and Optuna offers a robust, interpretable, and scalable approach for academic performance prediction. Future work should consider integrating additional behavioral and socio-economic variables to further refine the model's accuracy.

6. Declarations

6.1. Author Contributions

Conceptualization: I.P., R.I.D., and S.; Methodology: I.P.; Software: M.R.; Validation: M.R.; Formal Analysis: M.R.; Investigation: I.P., R.I.D., and S.; Resources: I.P., R.I.D., and S.; Data Curation: M.R.; Writing Original Draft Preparation: M.R.; Writing Review and Editing: I.P., R.I.D., and S.; Visualization: M.R.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors gratefully acknowledge the financial and institutional support provided by Universitas Multimedia Nusantara, including access to academic resources that supported this research.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable. This study used previously collected academic data and survey responses under the supervision of the university without any direct involvement from human participants.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] D. Lesjak, S. Natek, and F. Kohun, "Big data analytics in higher education," *Issues Inf. Syst.*, vol. 22, no. 4, pp. 320–333, 2021, doi: 10.48009/4_iis_2021_346-359.

[2] Á. Kocsis and G. Molnár, "Factors influencing academic performance and dropout rates in higher education," *Oxf. Rev. Educ.*, vol. 2024, no. Jan., pp. 1–12, 2024, doi: 10.1080/03054985.2024.2316616.

[3] T. Chamidy, M. A. Yaqin, and S. Suhartono, "The influence of internal and external factors on learning achievement," *Int. J. Lang. Lit. Media Stud.*, vol. 2022, no. Jan., pp. 562–573, 2022, doi: 10.2991/978-2-38476-002-2_53.

[4] S. S. Nartgun and E. Kennedy, "Learning analytics in decision making process in higher education institution," *Educ. Inf. Soc.*, vol. 2025, no. Apr., pp. 1–12, 2025, doi: 10.2991/978-94-6463-686-4_35.

[5] P. Saikia and K. Saha, "Factors affecting academic achievement: a systematic literature review," *Int. J. Soc. Sci. Res.*, vol. 2, no. 4, pp. 395–403, 2025, doi: 10.70558/IJSSR.2025.v2.i4.30510.

[6] J. H. Guanin-Fajardo, J. Guaña-Moya, and J. Casillas, "Predicting academic success of college students using machine learning techniques," *Data*, vol. 9, no. 4, pp. 1–12, 2024, doi: 10.3390/data9040060.

[7] E. Staneviciene, D. Gudoniene, V. Punys, and A. Kukstys, "A case study on the data mining-based prediction of students' performance for effective and sustainable e-learning," *Sustainability*, vol. 16, no. 23, pp. 1–12, 2024, doi: 10.3390/su162310442.

[8] O. K. Oyedotun, S. N. Tackie, E. O. Olaniyi, and A. Khashman, "Data mining of students' performance: Turkish students as a case study," *Int. J. Intell. Syst. Appl.*, vol. 7, no. 9, pp. 20–27, 2015, doi: 10.5815/ijisa.2015.09.03.

[9] A. Siddique, A. Jan, F. Majeed, A. I. Qahmash, N. N. Quadri, and M. O. A. Wahab, "Predicting academic performance using an efficient model based on fusion of classifiers," *Appl. Sci.*, vol. 11, no. 24, pp. 1–12, 2021, doi: 10.3390/app112411845.

[10] R. A. Yauri, H. U. Suru, J. Afrifa, and H. G. Moses, "A machine learning approach in predicting student's academic performance using artificial neural network," *J. Comput. Cogn. Eng.*, vol. 3, no. 2, pp. 203–212, 2024, doi: 10.47852/bonviewJCCE3202470.

[11] F. Kammüller and D. Satija, "Explanation of student attendance AI prediction with the Isabelle infrastructure framework," *Information*, vol. 14, no. 8, pp. 1–12, 2023, doi: 10.3390/info14080453.

[12] J. A. Ilemobayo, "Hyperparameter tuning in machine learning: a comprehensive review," *J. Eng. Res. Rep.*, vol. 26, no. 6, pp. 388–395, 2024, doi: 10.9734/jerr/2024/v26i61188.

[13] D. P. Prabha and C. V. Priscilla, "Probabilistic XGBoost threshold classification with autoencoder for credit card fraud detection," *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 11, no. Jan., pp. 528–537, 2023, doi: 10.17762/ijritcc.v11i8s.7234.

[14] A. Villar and C. R. V. de Andrade, "Supervised machine learning algorithms for predicting student dropout and academic success: a comparative study," *Discover Artif. Intell.*, vol. 4, no. 1, pp. 1–12, 2024, doi: 10.1007/s44163-023-00079-z.

[15] N. Y. Ling, T. T. Tin, T. C. Keat, U. F. Khattak, and M. A. Almaiah, "Educational big data analytics: machine learning based academic performance predictive modelling," *Pak. J. Life Soc. Sci.*, vol. 22, no. 2, pp. 7442–7477, 2024, doi: 10.57239/PJLSS-2024-22.2.00562.

[16] A. M. Shimaoka, R. C. Ferreira, and A. Goldman, "The evolution of CRISP-DM for data science: methods, processes and frameworks," *SBC Rev. Comput. Sci.*, vol. 4, no. 1, pp. 28–43, Oct. 2024, doi: 10.5753/reviews.2024.3757.

[17] G. Wallin, "An introduction to R and Python for data analysis: a side-by-side approach," *Am. Stat.*, vol. 78, no. 2, pp. 1–12, 2024, doi: 10.1080/00031305.2024.2320949.

[18] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Comput. Sci.*, vol. 7, no. Jul., pp. 1–24, 2021, doi: 10.7717/PEERJ-CS.623.

[19] T. O. Hodson, "Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not," *Geosci. Model Dev.*, vol. 2022, no. Jul., pp. 1–15, 2022, doi: 10.5194/gmd-15-5481-2022.

[20] R. H. Bemthuis, R. R. Govers, and A. Asadi, "A CRISP-DM-based methodology for assessing agent-based simulation models using process mining," *J. Simul.*, vol. 2025, no. Jun., pp. 1–22, 2025, doi: 10.1080/17477778.2025.2508245.

[21] B. K. Yousafzai, "Student performulator: student academic performance using hybrid deep neural network," *Sustainability*, vol. 13, no. 17, pp. 1–12, 2021, doi: 10.3390/su13179775.

[22] E. D. Lestari, "Explaining students' digital entrepreneurial behavior: the role of social media adoption in an integrated TPB–UTAUT model," *J. Appl. Data Sci.*, vol. 6, no. 4, pp. 2858–2875, Dec. 2025, doi: 10.47738/jads.v6i4.965.

[23] F. Nyimbili and L. Nyimbili, "Types of purposive sampling techniques with their examples and application in qualitative research studies," *Br. J. Multidiscip. Adv. Stud.*, vol. 5, no. 1, pp. 90–99, Feb. 2024, doi: 10.37745/bjmas.2022.0419.

[24] J. A. Solano, D. J. Lancheros Cuesta, S. F. Umaña Ibáñez, and J. R. Coronado-Hernández, "Predictive models assessment based on CRISP-DM methodology for students performance in Colombia - Saber 11 test," *Procedia Comput. Sci.*, vol. 2021, no. Jan., pp. 512–517, 2021, doi: 10.1016/j.procs.2021.12.278.

[25] A. B. Wiratman and W. Wella, "Personalized learning models using decision tree and random forest algorithms in telecommunication company," *Int. J. Inform. Vis.*, vol. 8, no. 1, pp. 318–325, 2024, doi: 10.62527/joiv.8.1.1905.

[26] D. Jacob and R. Henriques, "Educational data mining to predict bachelors students' success," *Emerg. Sci. J.*, vol. 7, no. Special Issue 2, pp. 159–171, 2023, doi: 10.28991/ESJ-2023-SIED2-013.

[27] J. Wu, X. Y. Chen, H. Zhang, L. D. Xiong, H. Lei, and S. H. Deng, "Hyperparameter optimization for machine learning models based on Bayesian optimization," *J. Electron. Sci. Technol.*, vol. 17, no. 1, pp. 26–40, Mar. 2019, doi: 10.11989/JEST.1674-862X.80904120.