JURTEKSI (Jurnal Teknologi dan Sistem Informasi) ISSN 2407-1811 (Print)
Vol. XII No 1, Desember 2025, hlm. 29 – 36 ISSN 2550-0201 (Online)
DOI: http://dx.doi.org/ 10.33330/jurteksi.v12i1.4160
Available online at http://jurnal.stmikroyal.ac.id/index.php/jurteksi

IMPLEMENTATION OF RANDOM FOREST CLASSIFIER FOR STUDENT GRADUATION CLASSIFICATION

Bazil Zaidan Putra[1*], Ika Nur Fajri[1], Agung Nugroho[1]
[1]Information Systems, Amikom University Yogyakarta
email: *bazilzputra@students.amikom.ac.id

Abstract: Higher education plays an essential role in improving human resource quality, one of which is through the institution’s ability to monitor and predict student graduation outcomes. This study does not focus on a specific university but utilizes the publicly available Students Performance in Exams dataset from Kaggle, consisting of 1,000 student records containing mathematics, reading, and writing scores, along with demographic attributes such as gender, parental education level, lunch type, and test preparation participation. The data were processed through a feature engineering stage by adding an *average score* variable as an early indicator of graduation status. A predictive model was developed using the Random Forest Classifier, achieving an accuracy of 99%. The final model was integrated into a Streamlit-based web application to provide an accessible tool for academic stakeholders. The results indicate that the proposed model can serve as an effective decision-support tool for early evaluation of students’ likelihood of graduation.

Keywords: prediction; random forest classifier, streamlit, student graduation.

Abstrak: Pendidikan tinggi memegang peran penting dalam peningkatan kualitas sumber daya manusia, salah satunya melalui kemampuan institusi dalam memantau dan memprediksi tingkat kelulusan mahasiswa. Penelitian ini tidak berfokus pada perguruan tinggi tertentu, melainkan menggunakan dataset publik Students Performance in Exams dari Kaggle yang berisi 1.000 data mahasiswa, terdiri atas nilai matematika, membaca, menulis, serta atribut demografis seperti gender, tingkat pendidikan orang tua, jenis makan siang, dan partisipasi kursus persiapan. Data diolah melalui tahap *feature engineering* dengan menambahkan variabel *average score* sebagai indikator awal kelulusan. Model prediksi dibangun menggunakan algoritma Random Forest Classifier, yang menghasilkan tingkat akurasi sebesar 99%. Model ini kemudian diimplementasikan ke dalam aplikasi web berbasis Streamlit untuk memberikan layanan prediksi yang mudah diakses oleh pihak akademik. Hasil penelitian menunjukkan bahwa model mampu digunakan sebagai alat pendukung keputusan untuk melakukan evaluasi dini terhadap potensi kelulusan mahasiswa.

Kata kunci: kelulusan mahasiswa; prediksi; random forest classifier; streamlit.

INTRODUCTION

Higher education plays an essential role in supporting human and economic development, and one indicator of its quality is the ability of students to complete their studies on time. Although the Gross Participation Rate (APK) of Higher Education in Indonesia reached 31.45% in 2023, this figure remains relatively low and is influenced by factors such as educational cost and limited access. Moreover, only about 10.20% of individuals aged 15 years and above have completed university-level education. Delays or failure to graduate not only increase students' financial burden but also affect institutional efficiency and reputation.

In recent years, Machine Learning (ML) has been increasingly utilized to support academic prediction, with Random Forest identified as one of the most effective algorithms for modeling student performance and graduation likelihood. Several studies demonstrate that Random Forest performs well in predicting academic outcomes across different educational settings, including Indonesian public high schools [1], graduation prediction tasks involving demographic and academic attributes[2], and university-level analyses where it outperforms other classification methods[3]. Factors that commonly contribute to graduation prediction include exam scores, GPA, demographic characteristics, and behavioral aspects such as course participation or preparatory class attendance[4], [5], while preprocessing steps like handling missing values, encoding categorical variables, and engineering features play an important role in improving prediction accuracy.

Despite its potential, many previous studies rely on institution-specific datasets and lack practical implementation, which limits generalizability and real-world applicability. To address these gaps, this study develops a Random Forest–based model to predict student graduation using academic test scores and demographic attributes, supported by structured preprocessing and comprehensive evaluation through accuracy, precision, recall, F1-score, and confusion matrix. The resulting model is implemented into a Streamlit-based web application to provide a practical tool for early monitoring and intervention by academic institutions, with a targeted accuracy of at least 90%.

METHOD

This research method consists of several stages: (1) data collection and preparation, (2) pre-processing and feature *engineering*, (3) data sharing, (4) model training using Random Forest, (5) model evaluation, and (6) application implementation.

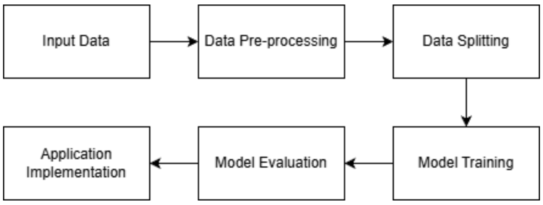


Figure 1. Research Flow Diagram

This research is carried out with several main stages, namely, data input, data pre-processing, data sharing, random forest model training, model evaluation, and finally application implementation.

Datasets and Data Collection

This study uses the publicly available Students Performance in Exams dataset on Kaggle with the URL: Students Performance in Exams.

The data includes math, reading, and writing test scores, as well as category attributes such as gender, race/ethnicity, parental education level, type of lunch, and exam prep courses. This dataset was chosen because it can represent the academic situation of students, both from a cognitive perspective and demographic factors.

Pre-Processing and Feature Engineering

The pre-processing stage is carried out to ensure the quality of the data. The missing values check is performed using the function df.isnull().sum(). Blank values are populated using mean (numerical) and mode (category) strategies, while duplicate data is deleted using df.drop_duplicates(inplace=True). Value validation is also performed to ensure a score range of 0–100.

If necessary, normalization can be applied using the z-score formula:

$$x' = \frac{x - \mu}{\sigma} \quad (1)$$

This formula is used to standardize the value of a feature by subtracting the mean value (μ) and dividing it by the standard deviation (σ). The normalization results make the data at the same scale so that the feature doesn't have too far a range of values, but in Random Forest this step is optional because the model is not sensitive to the difference in the scale of the feature.

Feature engineering is done by adding columns average_score using:

$$average\_score = \frac{math + reading + writing}{3} \quad (2)$$

Next, the approval label is created using the threshold:

$$graduation\_status = \begin{cases} "Passed",\ average\_score \geq 60 \\ "Not\ Passed",\ average\_score < 60 \end{cases} \quad (3)$$

All category features (gender, race/ethnicity, parental education, lunch, test preparation course) are converted into numerical values using Label Encoding so that they can be processed by the model. Some academic research has also shown that pre-processing techniques like this improve the performance of predictive models [6]

If class imbalances are found (e.g. the number of "Not Passed" is much smaller), oversampling methods such as SMOTE or undersampling can be applied to maintain a balance of class distribution, as is also done in similar studies.

Data Splitting

Once the data is clean and ready to use, the dataset is divided into 80% training data and 20% test data to test the model's generalization capabilities. This division is widely used in academic prediction research because it provides a good balance between learning capacity and independent validation.[7]

Model Random Forest Classifier

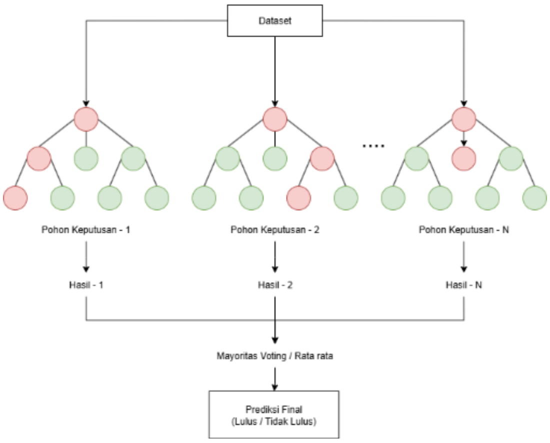


Figure 2. Random Forest Architecture

The model used in this study is the Random Forest Classifier, which is an ensemble of several decision trees built using the bootstrap sampling technique. Each tree provides a prediction, and the final result is determined based on the majority voting system. Random Forest was chosen because it is robust against overfitting and capable of handling complex data[8], [9].

The final prediction is given by the formula:

$$\hat{y} = mode\{h_1(x), h_2(x), \ldots, h_T(x)\}(4)$$

This formula shows that the final prediction of the Random Forest is obtained from a majority of votes (mode) of the entire decision tree, $h_1, h_2, \ldots, h_T x$. Each tree provides a prediction of the input data, then the class that appears most often becomes the model's prediction result. Split nodes based on Gini impurity i.e. Gini impurity Criterion ***G*** is used to select the best split:

$$G = 1 - \sum_{i=1}^{c} p_i^2 \ (5)$$

where p_i is the proportion of class i on the current node, and C is the sum of the class (here 2: "Passed" & "Not Passed").

Some important parameters are n_estimators(T) is the number of trees, max_features (mtry) is the number of randomly selected features in each split. (default: for classification, with $\sqrt{M}$ = total number of features), and max_depth, min_samples_split, min_samples_leaf = node depth and size limitations to avoid overfitting.

Each tree is built with ~2/3 of a bootstrap sample; The remaining ~1/3 (OOB) is used as internal validation data. OOB performance can provide error estimation without the need for a separate validation dataset.[8]

Model Evaluation

Performance evaluation was carried out using several metrics, namely accuracy, precision, recall, F1-score, and confusion matrix. The formulas used include:

Table 1. Confusion Matrix

	Positive Predictions	Negative Prediction
Positive Actuals	True Posotive	False Positive
Negative Actuals	False Negative	True Negative

The evaluation formula used is:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \ (6)$$

$$Precision = \frac{TP}{TP+FP} \ (7)$$

$$Recall = \frac{TP}{TP+FN} \ (8)$$

$$F1-Score\ = 2\ x\ \frac{Precision\ x\ Recall}{Precision+Recall} \ (9)$$

The use of these various metrics allows for a more comprehensive evaluation, including in the case of class imbalances.[7]

Application Implementation

The trained model is then stored in .sav format and integrated into a Streamlit-based web application. Users can enter student data and receive graduation predictions interactively. This approach makes it easier for academic institutions to access predictive systems without the need for in-depth technical understanding, as is also done in similar research[6].

To enhance usability, the application includes basic input validation and a clear presentation of prediction results. The system can also be expanded to display supporting information such as probability estimates or feature importance, allowing it to function not only as a prediction tool but also as an early academic monitoring aid for identifying students who may need further attention.

RESULT AND DISCUSSION

Input Data

The study used a dataset (Students Performance in Exams) from Kaggle containing 1,000 data with eight main features, namely gender, race/ethnicity, parental education level, type of lunch, participation in preparation courses, and three test scores (mathematics, reading, writing).[10], [11]

Table 2. Initial Data Snippet

Gender	Race/Ethnicity	Parental Level of Education	Lunch	Test Preparation Course	Math Score	Reading Score	Writing Score
Female	Group D	Some College	Standard	Completed	59	70	78
Male	Group D	Associate's Degree	Standard	None	96	93	87
Female	Group D	Some College	Free/Reduced	None	57	76	77
Male	Group B	Some College	Free/Reduced	None	70	70	63
Female	Group D	Associate's Degree	Standard	None	83	85	86

From the snippet, it can be seen that the dataset consists of a combination of categorical and numerical features. These characteristics are in line with previous research that used test score data and demographic attributes to predict academic performance.

Data Pre-processing

The initial stage of pre-processing is done by forming new features average_score using:

$$average\_score = \frac{math + reading + writing}{3} \quad (10)$$

This feature is used as the basis for determining the graduation status (≥60 = *Pass*, <60 = *Not Pass*).

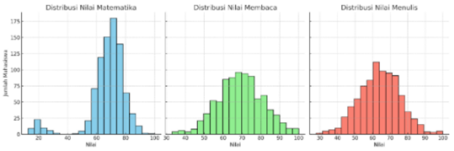


Figure 3. Visualization of Student Value Distribution

Shows that most scores are in the range of 50–80 with a *fairly even* distribution of Pass *and* Not Pass categories.

Table 3. Category Variable Encoding Label Results

Gender	Lunch	Test Preparation Course
0	1	0
1	0	1
0	1	0
1	0	1
0	1	0

Some numerical features are standardized when needed to equalize scale, although the Random Forest model is not sensitive to scale differences. Examination of the class distribution showed a small imbalance (*Pass classes* dominated 70–75%), but it was not significant enough to require SMOTE. These findings are consistent with the literature that assesses that simple pre-processing and encoding are adequate for the academic Random Forest model [6].

Data Splitting

After pre-processing, the data is divided into 80% training and 20% testing. This proportion is commonly used in predictive research because it provides a considerable amount of training data while also providing completely new test data.

Table 4. Data Distribution in Training and Testing Sets

Dataset	Total	Pass	Not Pass
Training Set	800	650	150
Testing Set	200	160	40
Total	1000	810	190

The similar class distribution between training and testing helps maintain the generalization of the model. This proportional data sharing is also widely used in Random Forest-based academic prediction research [12], [13]

Model Training

The model was trained using a Random Forest Classifier with key parameters *n_estimators = 100* and *random_state = 42*. The final prediction is obtained through the majority voting mechanism:

$$\hat{y} = mode\{h_1(x), h_2(x), \dots, h_T(x)\} (11)$$

With is the prediction result of the first decision tree, and $h_i(x)$ $\boldsymbol{N}$ is the number of trees in the forest.

This ensemble approach has been shown to be effective in reducing overfitting and providing predictive stability to educational datasets, as supported by previous research. [14]

Model Evaluation

The evaluation of the model's performance is carried out using several metrics: accuracy, precision, recall, and F1-score.

Table 5. Model Performance Evaluation Results

Algoritma	Accuration(%)	Precisision(%)	Recall(%)	F1-Score(%)
Logistic Regression	86.5	85.2	84.9	85.0
Decision Tree	89.7	88.5	87.9	88.2
Random Forest	99	100	99	99

Table 6. Calculation of Confusion Matrix Random Forest Classifier

	Predictions Pass	Predictions Not Pass
Actual Pass	158	2
Actual Not Pass	0	40

From the table, it can be calculated that the model has a precision of 100% and a recall of 99% for the graduated class. The F1-score metric also shows a consistently high value of 99%. The formula used is as follows:

$$Precision = \frac{TP}{TP+FP} = \frac{158}{158+0} = 1{,}00 \ (12)$$

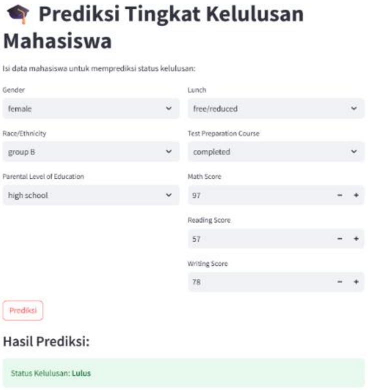
$$Recall = \frac{TP}{TP + FN} = \frac{158}{158 + 2} = 0{,}99 \ (13)$$

$$F1 - score \ = 2 \ x \ \frac{1,00 \ x \ 0,99}{1,00 + 0,99} \approx 0,99 \ (14)$$

These results suggest that the model is able to predict *the Pass* and *Not Pass* categories with minimal errors, consistent with studies reporting Random Forest's high performance on academic predictions. [15]

Application Implementation

The final model is saved in .sav format and integrated into Streamlit-based applications. Users can enter student data such as gender, race/ethnicity, parental education level, lunch type, prep course status, and math, reading, and writing scores.



🎓 Prediksi Tingkat Kelulusan Mahasiswa

Isi data mahasiswa untuk memprediksi status kelulusan:

Gender: female

Lunch: free/reduced

Race/Ethnicity: group B

Test Preparation Course: completed

Parental Level of Education: high school

Math Score: 97

Reading Score: 57

Writing Score: 78

Hasil Prediksi:

Status Kelulusan: **Lulus**

Figure 4. App Display

Once the data is entered, the system displays the graduation status based on the Random Forest model. The implementation of this application ensures that the research provides practical outputs that can be used as a *decision support system*, as recommended in related studies [1].

CONCLUSION

The results showed that the Random Forest Classifier algorithm was able to predict student graduation very well based on test scores and demographic attributes, resulting in a high accuracy of 99% with consistent precision, recall, and F1-score. Math scores were the most influential factors, followed by reading and writing, while demographic variables made an additional contribution with a smaller influence. The 80:20 data split shows the model's ability to generalize well to new data.

This study still has limitations because it only uses academic data and does not include non-academic factors such as attendance, learning motivation, or involvement in campus activities, as well as the imbalance of *Not Passing* classes that can affect the sensitivity of the model. Further research is suggested to add non-academic variables, apply classroom balancing techniques such as SMOTE, explore other algorithms such as XGBoost or Gradient Boosting, and test models in real environments to optimize their application.

BIBLIOGRAPHY

[1] R. Andriani Saputri and L. Rosnita, "A Random Forest-Based Predictive Model for Student Academic Performance: A Case Study in Indonesian Public High Schools," 2025. [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC

[2] F. Riskiyono and D. Mahdiana, "Implementation of Random Forest Algorithm for Graduation Prediction," *sinkron*, vol. 8, no. 3, pp. 1662–1670, Jul. 2024, doi: 10.33395/sinkron.v8i3.13750.

[3] I. Made and B. Adnyana, "PENERAPAN TEKNIK KLASIFIKASI UNTUK PREDIKSI KELULUSAN MAHASISWA

BERDASARKAN NILAI AKADEMIK.”

[4] M. Putra and Erwin Harahap, “Machine Learning pada Prediksi Kelulusan Mahasiswa Menggunakan Algoritma Random Forest,” *Jurnal Riset Matematika*, pp. 127–136, Dec. 2024, doi: 10.29313/jrm.v4i2.5102.

[5] A. Rahman, D. Mahdiana, and A. Fauzi, “Predicting Student On-Time Graduation Using Particle Swarm Optimization and Random Forest Algorithms,” *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 8, no. 1, p. 161, Feb. 2025, doi: 10.24014/ijaidm.v8i1.33577.

[6] S. A. A. Balabied and H. F. Eid, “Utilizing random forest algorithm for early detection of academic underperformance in open learning environments,” *PeerJ Comput Sci*, vol. 9, 2023, doi: 10.7717/peerj-cs.1708.

[7] R. Bakri, N. P. Astuti, and A. S. Ahmar, “Evaluating Random Forest Algorithm in Educational Data Mining: Optimizing Graduation on-time prediction using Imbalance Methods,” *ARRUS Journal of Social Sciences and Humanities*, vol. 4, no. 1, pp. 108–116, Feb. 2024, doi: 10.35877/soshum2449.

[8] D. Khairy, N. Alharbi, M. A. Amasha, M. F. Areed, S. Alkhalaf, and R. A. Abougalala, “Prediction of student exam performance using data mining classification algorithms,” *Educ Inf Technol (Dordr)*, vol. 29, no. 16, pp. 21621–21645, Nov. 2024, doi: 10.1007/s10639-024-12619-w.

[9] D. Doz, M. Cotič, and D. Felda, “Random Forest Regression in Predicting Students’ Achievements and Fuzzy Grades,” *Mathematics*, vol. 11, no. 19, Oct. 2023, doi: 10.3390/math11194129.

[10] M. Yağcı, “Educational data mining: prediction of students’ academic performance using machine learning algorithms,” *Smart Learning Environments*, vol. 9, no. 1, Dec. 2022, doi: 10.1186/s40561-022-00192-z.

[11] P. Matt, B. Holtman, E. Muja, and X. Li, “Student Performance on Exams.”

[12] S. Kumar Ghosh and F. Janan, “Prediction of Student’s Performance Using Random Forest Classifier.”

[13] C. Ma, “Improving the Prediction of Student Performance by Integrating a Random Forest Classifier with Meta-Heuristic Optimization Algorithms,” 2024. [Online]. Available: www.ijacsa.thesai.org

[14] M. Gusnina, Wiharto, and U. Salamah, “Student Performance Prediction in Sebelas Maret University Based on the Random Forest Algorithm,” *Ingenierie des Systemes d’Information*, vol. 27, no. 3, pp. 495–501, Jun. 2022, doi: 10.18280/isi.270317.

[15] F. A. S. Wibowo *et al.*, “IMPACT OF FEATURE SELECTION ON DECISION TREE AND RANDOM FOREST FOR CLASSIFYING STUDENT STUDY SUCCESS,” *Barekeng*, vol. 19, no. 3, pp. 2083–2096, Jul. 2025, doi: 10.30598/barekengvol19iss3pp2083-2096.