Vol. 8, No. 3, June 2026

Journal of Information Systems and Informatics

Published By Asosiasi Doktor Sistem Informasi Indonesia

An Empirical Comparison of C4.5, Naive Bayes, and KNN for Scholarship Selection

Burham Isnanto[1], Rahmat Sulaiman[2]

[1]Faculty of Economy and Business, ISB Atma Luhur, Pangkalpinang, Bangka Belitung, Indonesia

[2]Faculty of Information Technology, ISB Atma Luhur, Pangkalpinang, Bangka Belitung, Indonesia

Received:
October 21, 2025
Revised:
May 10, 2026
Accepted:
May 30, 2026
Published:
June 22, 2026

Corresponding Author:
Author Name*:
Burham Isnanto
Email*:
burham@atmaluhur.ac.id

DOI:
10.63158/journalisi.v8i3.1617

Abstract. Scholarship selection is a critical process in higher education that requires objective, fair, and efficient evaluation of applicants based on academic and socio-economic criteria. However, manual assessment methods are often vulnerable to bias, inconsistency, and administrative inefficiencies, which may affect the transparency and quality of decision-making. This study compares the performance of three supervised machine learning algorithms—C4.5 Decision Tree, Naive Bayes, and K-Nearest Neighbor (KNN)—for scholarship recipient classification. The dataset consisted of 1,500 student records obtained from the KelasAI repository and included ten predictor attributes, namely Grade Point Average, Parental Income, Academic Semester, Family Dependents, Organizational Involvement, Academic Achievement, Regional Origin, Scholarship Type, National Examination Score, and Economic Status. The target variable was categorized into Accepted and Rejected classes. Experiments were conducted using RapidMiner Studio with 10-fold stratified cross-validation to ensure reliable model evaluation. The results showed that Naive Bayes achieved the best performance, with 81.6% accuracy, 81.8% precision, and 81.3% recall, outperforming C4.5 and KNN. These findings demonstrate the potential of machine learning to support more transparent and data-driven scholarship selection processes.

Keywords: Scholarship Classification, Machine Learning, Comparative Benchmarking, Cross-Validation, Student Data Mining

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

1. INTRODUCTION

The rapid growth of digital data in higher education has created significant opportunities for applying machine learning (ML) to improve administrative and decision-making processes [1]. One important area that requires more objective and scalable decision support is scholarship allocation. Scholarship selection directly affects student welfare, institutional fairness, and academic achievement because it determines which students receive financial support to continue their studies [2]. However, in many higher education institutions, scholarship recipient selection is still commonly conducted through manual assessment, rule-based scoring, and subjective judgment. These conventional approaches may produce inconsistent decisions, limited scalability, and potential bias, especially when the number of applicants and selection criteria increase [3].

Educational Data Mining (EDM) offers a data-driven approach to address these limitations by applying computational techniques to analyze academic and socio-economic student data for institutional decision-making [4]. In this context, classification algorithms are particularly relevant because scholarship selection can be formulated as a prediction task that maps student attributes into predefined decision categories, such as eligible or not eligible for scholarship support [5]. Among supervised classification methods, C4.5 Decision Tree, Naive Bayes, and K-Nearest Neighbor (KNN) are frequently used in educational prediction tasks [6]. C4.5 is widely valued for its interpretability and ability to process continuous attributes through information gain ratio-based splitting. Naive Bayes uses probabilistic inference based on Bayes' theorem and can perform effectively even with its conditional independence assumption [7]. Meanwhile, KNN is a non-parametric algorithm that classifies data based on similarity patterns in the feature space [8].

Although these algorithms have been widely applied in educational prediction, several gaps remain. First, many studies focus on general academic performance prediction rather than scholarship recipient classification as a specific institutional decision problem [2], [3], [5]. Second, existing scholarship-related studies often evaluate only one or two algorithms, making it difficult to determine which model is more suitable under the same dataset and experimental conditions [14]. Third, comparative benchmarking studies that specifically analyze C4.5, Naive Bayes, and KNN on scholarship selection datasets are still

limited [9]. This limitation is particularly relevant in Indonesian higher education, where scholarship programs play an important role in supporting students financially and improving access to education [10].

Previous studies have demonstrated the potential of machine learning for scholarship and academic prediction tasks. Khan et al. proposed a machine learning-based framework for scholarship selection using several classifiers, including SVM, Neural Networks, KNN, and C4.5, and reported that C4.5 achieved strong performance in predicting scholarship beneficiaries using a Pakistani university dataset [1]. Broader studies on academic outcome prediction also show that classification-based approaches, particularly decision trees and probabilistic classifiers, tend to outperform purely rule-based methods in binary prediction tasks [2]. A systematic review of predictive learning analytics from 2012 to 2022 further identified Naive Bayes and Decision Tree variants as frequently used classifiers, with cross-validation being the dominant evaluation method [3].

Other studies have confirmed the relevance of these algorithms across different educational and benchmark datasets. Early prediction models for student performance in higher education have shown that probabilistic models can perform competitively on multi-attribute student datasets [5]. Comparative analysis of Naive Bayes, KNN, Decision Tree, and SVM on UCI benchmark datasets also found that Naive Bayes achieved strong accuracy, particularly on datasets with categorical attributes [6]. In the Indonesian context, previous research comparing KNN and Naive Bayes for scholarship admission prediction reported that Naive Bayes performed better than KNN [14]. Similar comparative studies involving C4.5 and Naive Bayes have also been applied in social welfare classification [15], while RapidMiner-based experiments using C4.5 and Naive Bayes for student graduation prediction provide a methodological precedent for tool-based classification pipelines [13]. In addition, prior scholarship prediction studies show that algorithm performance may vary depending on socio-economic feature composition [19], and rule-oriented scholarship eligibility frameworks indicate that academic performance and economic status are often among the most discriminative factors in scholarship classification [8].

Based on these problems and research gaps, this study aims to conduct a comparative evaluation of C4.5 Decision Tree, Naive Bayes, and KNN for scholarship recipient

classification. The study uses a dataset of 1,500 student records consisting of ten heterogeneous features, including numerical and categorical attributes. All experiments are conducted using RapidMiner Studio with stratified cross-validation to preserve class distribution across folds and produce a more balanced evaluation. Model performance is evaluated using accuracy, precision, and recall to identify the most suitable classification algorithm for scholarship recipient prediction under the tested dataset and experimental setup.

2. METHOD

2.1. Research Workflow

Figure 1 illustrates the research workflow adopted in this study. The process begins with problem identification and literature review to establish the research objectives and theoretical foundation. Subsequently, the dataset is collected and preprocessed to ensure data quality and suitability for analysis. The prepared data are then used for algorithm implementation, followed by experimental setup and execution using the selected machine learning models. Finally, the results are analyzed and discussed to evaluate model performance, leading to the formulation of conclusions and recommendations.

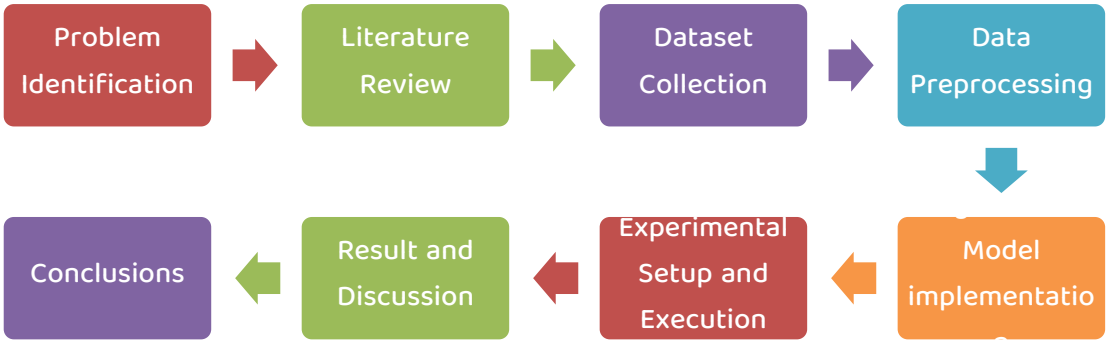

Figure 1. Research Workflow for Benchmarking Supervised Learning Algorithms in Automated Scholarship Selection

Figure 1 illustrates the overall research workflow used in this study for benchmarking machine learning algorithms in scholarship recipient classification. The workflow consists of eight sequential stages, starting from problem identification and ending with conclusions and implications. In the first stage, namely *Problem Identification*, the study

identifies the main issue in scholarship selection processes, which are still conducted manually and are often inefficient, time-consuming, and prone to human bias. This stage establishes the need for an automated and data-driven classification system. The second stage is *Literature Review*, where previous studies related to Educational Data Mining (EDM), machine learning classification, scholarship prediction systems, and benchmarking methods are reviewed. This stage provides the theoretical foundation and helps determine the algorithms and evaluation methods used in the research.

The third stage, *Dataset Collection*, involves obtaining a synthetic dataset from the KelasAI repository. The dataset consists of 1,501 student records with 10 feature attributes and one binary class label indicating Accepted or Rejected scholarship status. The fourth stage is *Data Preprocessing*. In this stage, missing values are handled using mean imputation for numerical attributes and mode imputation for categorical attributes. Categorical features are transformed using label encoding or one-hot encoding depending on the algorithm requirements, while numerical features are normalized using Min-Max normalization to improve model performance and maintain feature consistency. The fifth stage is *Algorithm Implementation*, where three supervised machine learning algorithms are implemented, namely C4.5 Decision Tree, Naive Bayes, and K-Nearest Neighbor (KNN). Each algorithm is configured using specific parameter settings in RapidMiner Studio to ensure fair comparative evaluation.

The sixth stage, *Experimental Setup & Execution*, describes the experimental process performed using RapidMiner Studio 12.0.005. A 10-fold stratified cross-validation approach is applied to evaluate model robustness while preserving class distribution across all folds. The models are evaluated using Accuracy, Precision, Recall, and F1-score metrics. The seventh stage is *Results & Evaluation*, where the performance of the three algorithms is compared based on the evaluation metrics. The experimental results indicate that Naive Bayes achieved the best classification performance compared to C4.5 and KNN. Finally, the eighth stage, *Conclusions & Implications*, summarizes the findings of the study and discusses the practical implications of implementing machine learning techniques for scholarship selection systems. This stage also provides recommendations for future research, including the use of ensemble learning and deep learning approaches to improve classification performance further.

2.2. Dataset Description

The dataset used in this study was synthetic data obtained from the KelasAI, an Indonesian educational data platform. The synthetic dataset was utilized to simulate scholarship selection scenarios and evaluate the performance of the proposed machine learning models. The dataset comprises 1,500 student records, each described by ten feature attributes and one binary class label. Table I presents the complete feature specification.

TABLE I. Dataset Feature Description

Feature	Description	Type
IPK	Grade Point Average (0.00-4.00)	Numeric
Penghasilan_Ortu	Parental Monthly Income (IDR)	Numeric
Semester	Current Academic Semester (1-8)	Numeric
Tanggungan	Number of Family Dependents	Numeric
Organisasi	Organizational Involvement	Categorical
Prestasi	Academic Achievement/Awards	Categorical
Asal_Daerah	Regional/Geographic Origin	Categorical
Jenis_Beasiswa	Scholarship Type Applied For	Categorical
Nilai_UN	National Examination Score	Numeric
Status_Ekonomi	Socio-Economic Status Level	Categorical
Status (Label)	Diterima / Tidak Diterima	Binary

2.3. Preprocessing

Prior to algorithm training, a systematic preprocessing pipeline was implemented to ensure data quality and algorithm compatibility [29]. Missing value imputation was performed using mode substitution for categorical attributes and mean substitution for numeric attributes. Categorical features were encoded using label encoding. Numeric features were normalized using min-max normalization to the range [0, 1] to mitigate scale sensitivity, particularly for the KNN algorithm [30]. No significant class imbalance was detected; therefore, oversampling techniques were not applied.

Missing values were identified and handled using [mean imputation for numerical features / mode imputation for categorical features] prior to encoding. The problem: Label encoding assigns ordinal integers (0, 1, 2…) to nominal categories, which introduces false distance ordering in KNN.

1) Naive Bayes - Label Encoding - Treats features probabilistically, encoding order does not affect posterior calculation.
2) C4.5 Decision Tree - Label Encoding - Splits on thresholds; ordinal distortion has minimal effect
3) KNN – One Hot encoding - Distance-based; nominal categories must not carry false ordinal weight

To ensure fairness in the comparative evaluation, categorical features were encoded differently per algorithm. Label encoding was applied for Naive Bayes and C4.5, while one-hot encoding was applied for KNN to prevent artificial distance distortion caused by arbitrary ordinal assignments in distance calculations.

KNN is sensitive to feature scale. Numerical features were normalized using Min-Max normalization to the range [0, 1] prior to KNN training, ensuring that features with larger magnitude ranges do not disproportionately influence Euclidean distance calculations. Min-Max formula as shown in Equation 1.

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

The dataset was partitioned using stratified k-fold cross-validation (k=10) to preserve class distribution across all folds. The class distribution of the dataset consists of 672 Accepted instances and 829 Rejected instances. Stratification ensures that both recipient and non-recipient classes are proportionally represented in each fold.

2.4. Modeling

C4.5 Decision Tree: The C4.5 algorithm constructs a decision tree by recursively partitioning the feature space using the information gain ratio criterion. Post-pruning via confidence factor (CF = 0.25) was applied to reduce overfitting. The algorithm natively handles both continuous and categorical attributes, making it well-suited for the mixed-type feature space of this dataset [13]. For the Decision Tree model in RapidMiner, the algorithm was configured using the Gain Ratio criterion with a maximum depth of 10. The

confidence level was set to 0.1, the minimal gain was 0.01, the minimal leaf size was 2, and the minimal size for split was 3. Additionally, the number of prepruning alternatives was set to 3. The Information Gain for an attribute $A$is calculated as:

$$Entropy(S) = \sum_{i=1}^{n} -Pi \log_2(Pi) \quad (2)$$

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{|Si|}{|S|} Entropy(Si) \quad (3)$$

Post-pruning is typically applied using a confidence factor (e.g., CF = 0.25) to reduce overfitting. C4.5 can naturally handle both continuous and categorical attributes.

Naive Bayes: The Gaussian Naive Bayes classifier was applied, assuming conditional independence between features given the class label. No additional hyperparameter tuning was required beyond the smoothing parameter, which was set to its default value (var_smoothing = 1e-9). The Naive Bayes classifier applies Bayes' theorem under the assumption of conditional independence between features given the class label. Laplace smoothing (alpha = 1) was applied to address zero-probability issues for categorical features [20]. Despite its independence assumption, Naive Bayes has demonstrated competitive performance in datasets with correlated features [7]. For the Naive Bayes model in RapidMiner, the default parameter settings were used with Laplace correction enabled to handle zero-frequency problems during probability estimation. Under the independence assumption, the likelihood is expressed as:

$$P(H \mid X) = \frac{P(X|Ci)P(Ci)}{P(X)} \quad (4)$$

$$P(H \mid X) = \frac{P(H|X)}{\sum_{i=1}^{n} P(Hi|X)} \; x \; P(H) \quad (5)$$

$$P(C \mid F1 \ldots Fn) = \frac{P(F1,\ldots Fn|C)}{P(F1\ldots Fn)} x \; P(C) \quad (6)$$

where $\alpha = 1$. Despite its simplifying assumption, Naive Bayes often performs competitively in various classification tasks.

K-Nearest Neighbor (KNN): KNN classifies instances by majority vote among the k nearest neighbors in feature space, measured using Euclidean distance. The optimal value of k was determined through grid search over k in {3, 5, 7, 9, 11}, with k = 5 yielding the best cross-validated performance [6]. One-hot encoding was applied for categorical attributes to avoid introducing artificial ordinal relationships, particularly for distance-based algorithms such as KNN. To avoid data leakage and ensure unbiased evaluation, hyperparameter optimization for KNN (specifically the value of k) was conducted using grid search strictly within the training folds of each cross-validation iteration. The optimal value of k was selected independently for each fold based on validation performance, and the test fold was never exposed during parameter tuning. For the k-Nearest Neighbors (k-NN) model in RapidMiner, the value of $k$was set to 5. The algorithm used mixed measures with Mixed Euclidean Distance as the distance metric, and weighted voting was enabled to improve classification performance based on neighbor proximity.

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{7}$$

where $N_k$represents the set of $k$nearest neighbors and $I$is the indicator function. The optimal value of $k$is typically determined through cross-validation or grid search.

2.5. Experimental Setup

All experiments were conducted in RapidMiner Studio 9.10. Model evaluation was performed using 10-fold stratified cross-validation to ensure unbiased performance estimation [30]. Performance metrics include Accuracy, Precision (weighted average), and Recall (weighted average), defined as shown in Equation 8 to 10.

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \tag{8}$$

$$\text{Precision} = TP / (TP + FP) \tag{9}$$

$$\text{Recall} = TP / (TP + FN) \tag{10}$$

where TP, TN, FP, and FN denote True Positives, True Negatives, False Positives, and False Negatives, respectivel [30].

3. RESULTS AND DISCUSSION

3.1. Performance Evaluation

Table 2 summarizes the classification performance of the three algorithms evaluated across all ten-fold cross-validation iterations. Results represent mean values across all folds. The performance evaluation of the proposed models was conducted using the Cross Validation operator in RapidMiner with the Performance Classification operator applied in the testing phase. A 10-fold cross-validation approach was employed to ensure reliable and unbiased model evaluation. In this process, the dataset was divided into 10 subsets, where 9 subsets were used for training and 1 subset was used for testing iteratively until all subsets had been evaluated. The evaluation metrics included Accuracy, Precision, Recall, and F1-score. Accuracy measures the overall proportion of correctly classified instances, while Precision evaluates the proportion of correctly predicted positive instances among all predicted positives. Recall measures the proportion of actual positive instances correctly identified by the model, and F1-score represents the harmonic mean of Precision and Recall to provide a balanced evaluation of classification performance. These metrics were automatically calculated using the Performance Classification operator for each fold, and the final results were obtained by averaging the performance values across all folds.

Table 2. Comparative Classification Performance

Model	Accuracy (%)	Precision (%)	Recall (%)
Naive Bayes	81.60	81.80	81.30
C4.5 Decision Tree	78.60	78.70	78.20
K-Nearest Neighbor	76.20	76.10	75.80

The table presents the comparative performance of three classification algorithms—C4.5 Decision Tree, Naive Bayes, and K-Nearest Neighbor (KNN)—based on accuracy, precision, and recall. Overall, Naive Bayes demonstrates the best performance among the three algorithms, achieving the highest accuracy of 81.60%, precision of 81.80%, and recall of 81.30%. This indicates that Naive Bayes provides the most balanced and reliable

classification results for this dataset, despite its simplifying assumption of feature independence.

The C4.5 Decision Tree shows moderate performance, with an accuracy of 78.60%, precision of 78.70%, and recall of 78.20%. These results suggest that C4.5 is capable of capturing patterns in the data reasonably well, particularly due to its ability to handle both categorical and continuous features. However, its performance is slightly lower than that of Naive Bayes. Meanwhile, K-Nearest Neighbor (KNN) yields the lowest performance among the three models, with an accuracy of 76.20%, precision of 76.10%, and recall of 75.80%. This may be due to its sensitivity to the choice of parameter $k$and the distribution of data in the feature space, which can affect its ability to generalize effectively.In conclusion, Naive Bayes outperforms both C4.5 and KNN across all evaluation metrics, making it the most suitable algorithm for this classification task based on the experimental results.

3.2. Discussion

The experimental results demonstrate clear performance differentiation among the three algorithms. The experimental results indicate that Naive Bayes outperformed the other evaluated algorithms on this dataset, with accuracy (81.6%), precision (81.8%), and recall (81.3%), representing a margin of approximately 3.9 percentage points over C4.5 and 6.3 percentage points over KNN in terms of accuracy.

Naive Bayes achieved the highest average performance among the evaluated models and can be attributed to several dataset-specific characteristics. The presence of multiple categorical features (Organisasi, Prestasi, Asal_Daerah, Jenis_Beasiswa, Status_Ekonomi) is particularly favorable for probabilistic classifiers, as Naive Bayes constructs conditional probability tables that capture feature-class associations without requiring complex splitting structures [14]. Furthermore, the Laplace smoothing mechanism effectively handles low-frequency categorical values [24].

C4.5 achieved moderate performance (accuracy: 78.6%), benefiting from its ability to handle mixed data types and construct interpretable decision paths [13]. The tree structure identified IPK (GPA) and Status_Ekonomi (Economic Status) as the most discriminative features, consistent with domain knowledge on scholarship criteria [8].

However, the algorithm's susceptibility to overfitting on complex feature interactions, even with post-pruning, may partially explain the performance gap relative to Naive Bayes [17].

KNN demonstrated the lowest performance (accuracy: 76.2%), attributable to the curse of dimensionality in a ten-dimensional feature space. Despite normalization, the Euclidean distance metric may inadequately represent semantic similarity between categorical features, reducing the quality of nearest-neighbor identification [6]. This limitation is consistent with findings from Kushartanto and Aldisa [14] who similarly reported KNN underperformance on scholarship datasets with mixed feature types. Notably, the consistency between precision and recall values across all three algorithms suggests balanced classification performance with respect to both positive (Diterima) and negative (Tidak Diterima) classes, indicating the absence of significant class imbalance bias in the evaluation [30].

The proposed experimental framework contributes a reproducible benchmarking pipeline for scholarship classification tasks. The framework is distinguished by: (1) algorithm-specific preprocessing to ensure encoding fairness, (2) nested cross-validation to prevent data leakage during hyperparameter tuning, and (3) a standardized three-metric evaluation protocol (accuracy, precision, recall) enabling direct and fair comparison across probabilistic, tree-based, and distance-based classifiers.

From a practical standpoint, the findings of this study have direct implications for higher education institutions seeking to automate scholarship selection processes. The Naive Bayes classifier's combination of high accuracy, computational efficiency, and interpretable probabilistic outputs makes it particularly suitable for deployment in resource-constrained institutional environments. The algorithm's transparency in expressing classification decisions as posterior probabilities enables administrators to understand and audit automated recommendations. In conclusion, Naive Bayes outperforms both C4.5 and KNN across all evaluation metrics, making it the most suitable algorithm for this classification task based on the experimental results, as shown in Figure 2.

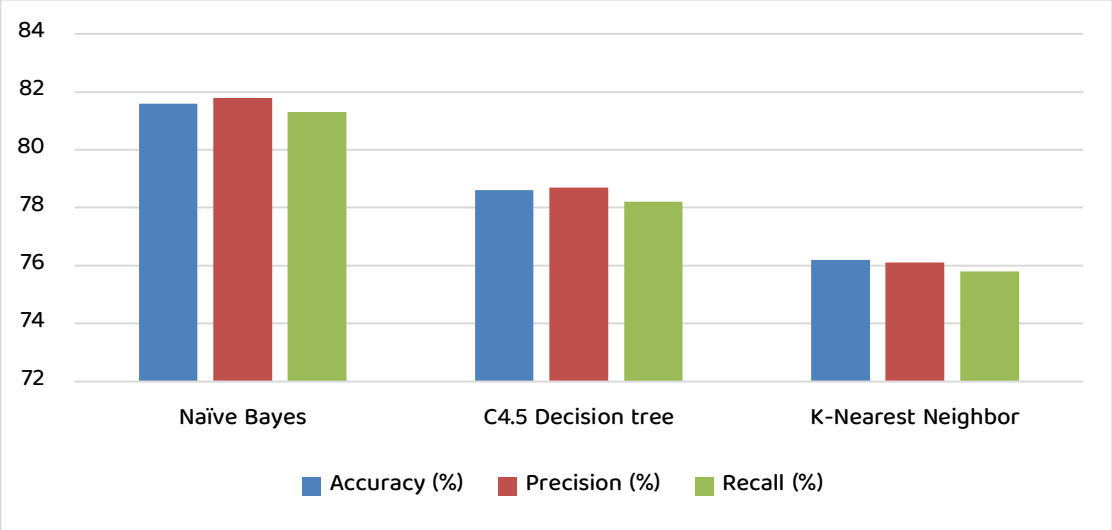

Figure 2. Classification Algorithm Performance on Scholarship Selection Dataset

Based on Figure 2, Naive Bayes leads across all metrics. With 81.6% accuracy, 81.8% precision, and 81.3% recall, it consistently outperforms the other two classifiers. Its probabilistic nature makes it well-suited for this kind of classification task, and the margins over its competitors are meaningful enough to make it the clear recommendation. C4.5 Decision Tree sits in the middle. Its scores (78.6% accuracy, 78.7% precision, 78.2% recall) are about 3 percentage points below Naive Bayes. The gap is modest, and its main practical advantage — producing human-readable decision rules — makes it a strong candidate when explainability or auditability is a priority for institutional stakeholders. KNN lags behind**.** At 76.2% accuracy, 76.1% precision, and 75.8% recall, it ranks last on every metric. Its weakness here likely stems from sensitivity to hyperparameter choices (especially *k*) and how the feature space is structured. That said, it's not dramatically worse — only about 5.4 percentage points behind Naive Bayes — and with better preprocessing or distance metric tuning, it could be competitive.

For higher education institutions operating in resource-constrained environments, Naive Bayes presents the most deployable option given its computational efficiency, probabilistic transparency, and superior classification performance on this dataset. C4.5 remains viable where decision auditability is prioritized over marginal accuracy gains One notable observation: all three models show very consistent behavior across the three metrics, meaning none of them trades off precision for recall in a lopsided way. This suggests the dataset is relatively balanced and the classifiers are stable.

4. CONCLUSION

This study conducted a comparative evaluation of three machine learning classification algorithms — Naive Bayes, C4.5 Decision Tree, and K-Nearest Neighbor (KNN) — for automated scholarship recipient classification. Experimental results obtained through 10-fold cross-validation demonstrated that Naive Bayes achieved the highest performance across all evaluation metrics, with an accuracy of 81.60%, precision of 81.80%, and recall of 81.30%, followed by C4.5 Decision Tree (accuracy: 78.60%, precision: 78.70%, recall: 78.20%), and KNN (accuracy: 76.20%, precision: 76.10%, recall: 75.80%). These findings suggest that Naive Bayes is the most suitable algorithm for this classification task, owing to its probabilistic transparency, computational efficiency, and robust handling of mixed feature types present in the scholarship dataset. However, it must be noted that these results are limited to the dataset used in this study, and broader institutional validation across different demographic contexts and larger applicant datasets is still needed before these findings can be generalized. Future work should explore ensemble methods, feature selection strategies, and deep learning approaches to further improve classification performance in scholarship selection systems. Future research directions include the exploration of ensemble methods and deep learning architectures to further improve predictive accuracy. Additionally, feature importance analysis, imbalanced class handling strategies, and external validation on multi-institutional datasets represent promising avenues for extending the generalizability of these findings.

REFERENCES

[1] H. U. Khan, F. V. Espiritu, and M. C. B. Natividad, "A new framework for scholarship predictor using a machine learning approach," *Intelligent Automation & Soft Computing*, vol. 39, no. 5, pp. 949–964, 2024. doi: 10.32604/iasc.2024.058466.

[2] P. Valdiviezo-Diaz and J. Chicaiza, "Prediction of academic outcomes using machine learning techniques: A survey of findings on higher education," *Communications in Computer and Information Science*, vol. 2049, pp. 218–232, 2024. doi: 10.1007/978-3-031-58956-0_16.

[3] N. Sghir, A. Adadi, and M. Lahmer, "Recent advances in predictive learning analytics: A decade systematic review (2012–2022)," *Education and Information Technologies*, vol. 28, no. 7, pp. 8299–8333, 2023. doi: 10.1007/s10639-022-11536-0.

[4] P. Nayak, S. Vaheed, S. Gupta, and N. Mohan, "Predicting students' academic performance by mining the educational data through machine learning-based classification model," *Education and Information Technologies*, vol. 28, no. 11, pp. 14611–14637, Nov. 2023, doi: 10.1007/s10639-023-11706-8.

[5] E. Alhazmi and A. Sheneamer, "Early predicting of students performance in higher education," *IEEE Access*, vol. 11, pp. 27579–27589, 2023. doi: 10.1109/ACCESS.2023.3258083.

[6] V. Sheth, P. Ramteke, V. Saxena, and A. Kumar, "A comparative analysis of machine learning classification algorithms for binary classification," *Procedia Computer Science*, vol. 215, pp. 422–431, 2022. doi: 10.1016/j.procs.2022.12.044.

[7] M. Yagci, "Educational data mining: Prediction of students' academic performance using machine learning algorithms," *Smart Learning Environments*, vol. 9, no. 1, p. 11, 2022. doi: 10.1186/s40561-022-00192-z.

[8] Y. Alshamaila, I. Al-Shourbaji, A. Alam *et al.*, "An intelligent rule-oriented framework for extracting key factors for grants scholarships in higher education," *International Journal of Data and Network Science*, vol. 8, no. 2, pp. 1325–1340, 2024. doi: 10.5267/j.ijdns.2023.11.002.

[9] H. Karalar, C. Kapucu, and H. Gurler, "Predicting students at risk of academic failure using ensemble model during pandemic in a distance learning system," *International Journal of Educational Technology in Higher Education*, vol. 18, no. 1, p. 63, 2021. doi: 10.1186/s41239-021-00300-y.

[10] B. Albreiki, N. Zaki, and H. Alashwal, "A systematic literature review of student performance prediction using machine learning techniques," *Education Sciences*, vol. 11, no. 9, p. 552, 2021. doi: 10.3390/educsci11090552.

[11] G. Brotosaputro, E. Helmud, and R. Sulaiman, "Comparative Accuracy of Prediction Classification Using Supervised Machine Learning," in *Proceedings of the 2025 7th International Conference on Cybernetics and Intelligent System (ICORIS)*, Mataram, Indonesia, 2025, pp. 1–6, doi: 10.1109/ICORIS67789.2025.11296063.

[12] R. Alamri and B. Alharbi, "Explainable student performance prediction models: A systematic review," *IEEE Access*, vol. 9, pp. 33132–33143, 2022. doi: 10.1109/ACCESS.2022.3061502.

[13] A. Tholib, M. N. F. Hidayat, S. Yono, R. Wulanningrum, and E. Daniati, "Comparison of C4.5 and Naive Bayes for predicting student graduation using machine learning algorithms," *International Journal of Engineering and Computer Science Applications (IJECSA)*, vol. 2, no. 2, pp. 71–78, 2023. doi: 10.30812/ijecsa.v2i2.3364.

[14] N. A. Kushartanto and R. T. Aldisa, "Data mining perbandingan algoritma K-Nearest Neighbor dan Naive Bayes dalam prediksi penerimaan beasiswa," *Journal of Computer System and Informatics (JoSYC)*, vol. 5, no. 1, pp. 196–207, 2023. doi: 10.47065/josyc.v5i1.4566.

[15] P. Ramadani, R. Fadillah, Q. Adawiyah, and B. R. Al Ghazali, "Perbandingan algoritma Naive Bayes, C4.5, dan K-Nearest Neighbor untuk klasifikasi kelayakan Program Keluarga Harapan," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 8, no. 2, pp. 311–319, 2024. doi: 10.29207/resti.v8i2.5812.

[16] E. F. Wati, E. S. Perangin-Angin, and L. Indriyani, "Comparison of Naive Bayes and C4.5 methods with Particle Swarm Optimization on customer loyalty classification," *IJISTECH (International Journal of Information System and Technology)*, vol. 8, no. 6, pp. 680–691, 2025. doi: 10.30645/ijistech.v8i6.382.

[17] V. Fitriyanti, G. Testiana, and C. E. Gunawan, "Klasifikasi predikat kelulusan mahasiswa menggunakan algoritma C4.5," *Jurnal Saintekom: Sains, Teknologi, Komputer dan Manajemen*, vol. 14, no. 2, pp. 217–232, 2024.

[18] F. Adiani, N. Fardiani, and F. Fitriyani, "Penerapan algoritma C4.5 untuk prediksi penerima beasiswa siswa berprestasi," *JIKA (Jurnal Informatika)*, vol. 8, no. 4, pp. 465–474, 2024. doi: 10.31000/jika.v8i4.12117.

[19] N. T. Haryati, E. S. Negara, and T. B. Kurniawan, "Klasifikasi pemberian beasiswa berprestasi menggunakan perbandingan tiga algoritma," *Jurnal TEKNOINFO*, vol. 17, no. 1, pp. 145–152, 2023. doi: 10.33365/jti.v17i1.2423.

[20] A. Anwarudin, W. Andriyani, B. P. DP, and D. Kristomo, "The prediction on the students' graduation timeliness using Naive Bayes classification and K-Nearest Neighbor," *Journal of Intelligent Software Systems*, vol. 1, no. 1, pp. 75–88, 2022. doi: 10.26798/jiss.v1i1.597.

[21] W. I. Kurniawan and J. Triloka, "Application of Naive Bayes classifiers for family risk identification and stunting intervention planning," *Journal of Applied Informatics and Computing*, vol. 9, no. 5, pp. 1156–1165, 2025. doi: 10.30871/jaic.v9i5.9143.

ISI Journal of Information Systems and Informatics
Published By Asosiasi Doktor Sistem Informasi Indonesia

[22] D. A. Shafiq, M. Marjani, R. A. A. Habeeb, and D. Asirvatham, "Student retention using educational data mining and predictive analytics: A systematic literature review," *IEEE Access*, vol. 10, pp. 72480–72503, 2022. doi: 10.1109/ACCESS.2022.3189214.

[23] M. B. Al-Zoubi, A. S. Al-Hashemi, and S. H. El-Gayar, "A review of educational data mining in higher education," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 5, pp. 458–467, 2021. doi: 10.14569/IJACSA.2021.0120652.

[24] S. Hussain and M. Q. Khan, "Student-Performulator: Predicting students' academic performance at secondary and intermediate level using machine learning," *Annals of Data Science*, vol. 10, no. 3, pp. 637–655, 2023. doi: 10.1007/s40745-021-00341-0.

[25] N. Aprilyani, I. Zulfa, and H. Syahputra, "Penerapan algoritma Decision Tree C4.5 untuk model penentuan penerima beasiswa Program Indonesia Pintar (PIP) studi kasus SMA Negeri 3 Timang Gajah," *Jurnal Teknik Informatika dan Elektro*, vol. 5, no. 1, pp. 23–34, 2022.

[26] B. Isnanto and R. Sulaiman, "Optimalisasi pembangunan desa: Prediksi kebutuhan intervensi ekonomi di Jawa Barat menggunakan algoritma machine learning," *Buffer Informatika*, vol. 12, no. 1, pp. 80–86, 2026.

[27] M. B. Alqahtani and E. Alqahtani, "Educational data mining and predictive modeling in the age of artificial intelligence: An in-depth analysis of research dynamics," *Computers*, vol. 14, no. 2, p. 68, 2025. doi: 10.3390/computers14020068.

[28] S. Berutu, H. Budiati, J. Jatmika, and F. Gulo, "Data preprocessing approach for machine learning-based sentiment classification," *Journal INFOTEL*, vol. 15, no. 4, pp. 317–325, 2023. doi: 10.20895/infotel.v15i4.1030.

[29] M. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 253–260, 2022. doi: 10.1016/j.gltp.2022.04.020.

[30] K. Vujovic, "Classification model evaluation metrics," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 6, pp. 599–606, 2021. doi: 10.14569/IJACSA.2021.0120670.