Vol. 8, No. 3, June 2026

ISI Journal of Information Systems and Informatics

Published By Asosiasi Doktor Sistem Informasi Indonesia

Predicting Student Performance to Support Adaptive Content Delivery: A Random Forest Approach

I Kadek Dwi Nuryana[1], Lintang Iqhtiar Dwi Mawarni[2]

[1,2]Information System Department, Engineering Faculty, Universitas Negeri Surabaya, Surabaya, Indonesia

Received:
October 13, 2025
Revised:
May 17, 2026
Accepted:
June 24, 2026
Published:
June 27, 2026

Corresponding Author:
Author Name*:
I Kadek Dwi Nuryana
Email*:
dwinuryana@unesa.ac.id

DOI:
10.63158/journalisi.v8i3.1663

Purpose: This study addresses the prediction-to-action gap in student performance analytics by proposing an interpretable framework that transforms predictive risk scores into adaptive content recommendations. Rather than only identifying at-risk students, the framework integrates performance prediction, interpretable rule extraction, and decision-support simulation to guide adaptive learning interventions. The study used the Open University Learning Analytics Dataset (OULAD), comprising 6,937 student records after filtering and preprocessing from the original 32,593 records. A Random Forest-based framework was adopted because of its interpretability and rule-extraction capability, although XGBoost achieved slightly higher predictive performance. The framework consists of three components: student performance prediction, interpretable decision rule extraction, and a decision-engine simulation for adaptive content recommendation. The predictive model achieved 87.22% accuracy and an AUC-ROC of 0.932. Rule extraction generated 20 human-readable rules with an average of 2.0 conditions per rule, an interpretability score of 1.000, and 81.6% fidelity to the full Random Forest model. The decision-engine simulation classified students by risk level and produced corresponding adaptive recommendations. An estimated Adaptation Gain metric indicated a potential 53.54% improvement in projected student success rates under conservative simulation assumptions. The proposed framework connects prediction with actionable recommendations to support educational decision-making, although real-world intervention validation remains necessary.

Keywords: Student performance prediction; Learning analytics; Adaptive content recommendation; Random Forest; Interpretable machine learning.

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

1. INTRODUCTION

The rapid expansion of online and blended learning environments has generated vast amounts of student interaction data stored in learning management systems. These digital footprints, including clickstream activity, assessment submissions, and forum participation, provide substantial opportunities for developing predictive models aimed at identifying at-risk students before academic failure occurs [1], [2], [3], [4]. Educational institutions increasingly recognize that early identification enables timely interventions, which in turn improve retention and learning outcomes [5], [6], [7]. However, a critical gap persists between making accurate predictions and translating those predictions into actionable pedagogical interventions [8], [9] This study defines the prediction-to-action gap as the disconnect between generating risk predictions and translating them into concrete pedagogical actions. Although predictive accuracy in learning analytics has improved substantially, the operationalization of prediction results into practical intervention strategies remains limited.

The Open University Learning Analytics Dataset (OULAD) has emerged as a well-known benchmark for evaluating machine learning effectiveness in predicting student performance [1], [10]. Existing studies can be categorized into four main streams. First, prediction accuracy studies: Borna et al. [11] compared several algorithms and found Random Forest to achieve the highest accuracy at 78.68% in predicting student withdrawal. Omarbekova et al. [12] developed a hybrid deep learning approach combining Bi-LSTM with temporal attention mechanisms, achieving an ROC-AUC of 0.95 and a weighted F1-score of 0.90. Rimal and Sharma [13] trained ensemble methods for multiclass grade prediction, finding that C-grade prediction was accurate at 97% while A-grade prediction remained challenging. Second, explainability studies: Guevara-Reyes et al. [14] addressed interpretability using SHAP-based techniques, demonstrating that five variables accounted for 72% of performance variability. Srivastava [16] employed an Explainable AI framework with SHAP and LIME to provide comprehensive prediction and determine significant factors. Third, early warning system studies: Akçapınar et al. [6], [7] developed early-warning systems using eBook interaction logs and LMS data. Taqatqeh and Agoyi [5] demonstrated the effectiveness of a complete prediction-intervention-evaluation cycle in higher education. Fourth, adaptive intervention studies: Abed [15]

proposed adaptive learning with attention-based knowledge tracing, while Banihashem et al. [16] examined the impacts of constructivist learning design on student engagement.

Despite these advances, a fundamental limitation persists across all four categories: most studies terminate at the prediction level. Borna et al. [11] reported accuracy metrics but did not discuss how predictions could inform content adaptation. Guevara-Reyes et al. [14] explicitly recognized this gap, stating that "most existing approaches are not interpretable and do not provide actionable insights to support decision-making." Furthermore, recent studies have highlighted challenges in explainability and the practical application of AI-driven educational interventions [8], [17]. While early warning systems [5], [6] identify at-risk students and recommendation systems suggest interventions, none provide an integrated framework that (a) predicts risk with high accuracy, (b) extracts human-readable rules for educator understanding, and (c) maps risk probabilities to specific adaptive content actions in a simulated decision engine.

Unlike conventional early-warning systems, which primarily identify students at risk and notify educators, the proposed framework extends the process by extracting interpretable decision rules and translating risk estimates into structured recommendation scenarios. Likewise, unlike recommendation-oriented studies that focus on suggesting interventions, this framework explicitly links prediction, explanation, and recommendation within a single workflow. Nevertheless, the recommendation component remains a simulation-based decision-support mechanism and has not yet been deployed as a fully operational adaptive learning system.

Therefore, this research intends to address the prediction-to-action gap by proposing an integrated framework that combines prediction with adaptive content delivery. The choice of Random Forest over XGBoost or deep learning methods (e.g., Bi-LSTM, attention-based networks) is justified by three considerations: (1) Random Forest provides inherent interpretability through tree structures and feature importance, unlike deep learning black-box models; (2) prior studies [11], [18] demonstrated that Random Forest achieves competitive or superior performance on OULAD compared to more complex models; and (3) the ensemble nature of Random Forest enables direct extraction of decision rules without requiring post-hoc explanation methods like SHAP or LIME, which have been shown to produce variable and sometimes inconsistent explanations [17].

The pedagogical basis for the proposed adaptive actions draws from three complementary perspectives. Vygotsky's Zone of Proximal Development (ZPD) [19] suggests that learning is most effective when instructional support is aligned with learners' current capabilities. Scaffolding theory [20] provides the mechanism for adjusting support intensity according to student needs, while differentiated instruction [21] emphasizes tailoring learning experiences to different learner profiles. Based on these principles, the decision engine assigns three levels of support: (a) Enrichment (low risk, $p < 0.3$), which provides advanced materials and extension activities for students likely to succeed independently [22]; (b) Light scaffolding (medium risk, $0.3 \leq p < 0.7$), which offers guided support and structured learning resources to promote self-regulated learning [23]; and (c) Full remediation (high risk, $p \geq 0.7$), which recommends intensive support and additional learning resources for students requiring substantial assistance. This tiered strategy reflects the principle that instructional support should be proportional to learner needs and risk levels [5], [15], [24].

To guide the evaluation, this study addresses the following research questions (RQs) :

1) RQ1: How accurately can a Random Forest model predict student performance (pass/fail) using OULAD data?
2) RQ2: What interpretable decision rules can be extracted from the Random Forest ensemble to help educators understand risk factors?
3) RQ3: How effectively can a decision engine map risk probabilities to adaptive content recommendations in terms of Risk Coverage and estimated Adaptation Gain?
4) RQ4: To what extent does the proposed framework differ from existing prediction-only and explainability-only approaches?

The objectives of this research are: (1) to build a predictive model based on Random Forest achieving ≥85% accuracy; (2) to extract human-readable decision rules with an average of ≤3 conditions per rule; (3) to develop a decision engine that simulates adaptive content recommendations based on risk thresholds; and (4) to propose and compute new evaluation metrics including Adaptation Gain (estimated improvement) and Risk Coverage (proportion of high-risk students identified).

The novelty of this study lies in integrating student performance prediction, interpretable rule extraction, and recommendation simulation within a unified learning analytics framework. Unlike prediction-focused studies [11], [12], [25] the proposed framework extends predictive outputs into recommendation scenarios. Unlike explainability-focused approaches that primarily rely on post-hoc interpretation [14], [17] this study extracts human-readable rules directly from the Random Forest ensemble. Furthermore, rather than functioning as a deployed adaptive learning system, the proposed decision engine serves as a simulation-based decision-support mechanism for educators, providing a decision-support simulation for educational planning.

2. METHODS

2.1 Research Flow

Figure 1 illustrates the research workflow, which includes data preparation, feature engineering, data preprocessing, model training, rule extraction, decision engine development, and evaluation [2], [26], [27].

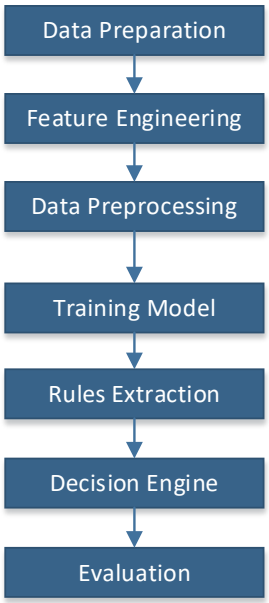

Figure 1. Research Workflow Diagram

2.2 Dataset

This study used the Open University Learning Analytics Dataset (OULAD), which contains demographic information and aggregated clickstream data of student interactions within

the Virtual Learning Environment (VLE) [1]. The dataset covers 22 courses, 32,593 students, assessment results, and daily summaries of VLE interactions (10,655,280 entries). Data were collected from seven modules across two presentations (2013J and 2014J). For computational efficiency, the data were filtered to two course modules (AAA and BBB, the most populated) and sampled to 200,000 interaction records. This resulted in 6,937 unique students with complete feature and target data [25], [26]. The selection of modules AAA and BBB was based on their having the largest student populations, which ensures sufficient sample size for model training and statistical power. However, this selection may limit generalizability to smaller courses or courses with different assessment structures. Future work should validate the framework across all 22 courses. Table 1 presents sample records from the preprocessed dataset, illustrating the structure of engagement, assessment, and demographic features used for prediction.

Table 1. Sample Records from Preprocessed Dataset (selected features only)

Student ID	total_clicks	avg_score	assessments_taken	pref_quiz	num_of_prev_attempts	std_clicks	studied_credits	final_result_binary
6516	272	61.8	5.0	0	0	6.66	60	1
11391	83	82.0	5.0	0	0	2.27	240	1
23629	12	82.5	4.0	3	2	0.89	60	0
23798	46	93.91	11.0	7	0	3.04	60	1
24734	29	46.8	5.0	0	0	0.67	60	1

2.3 Feature Engineering

A total of 27 predictive features were derived from four categories, following the methodology established by Kuzilek et al. [1] and extended by prior educational data mining studies [3], [4], [7]. Feature engineering was performed using Python pandas aggregation and transformation operations.

1) Engagement Features: From the StudentVLE table, the following metrics were calculated per student using pandas groupby operations: total_clicks (sum of all clicks), avg_clicks (mean clicks per interaction), std_clicks (standard deviation of click activity), interaction_count (total number of VLE interactions), and unique_days (number of unique days with activity). Engagement-related variables have consistently demonstrated strong predictive power in student

success prediction studies [3], [4], [9]. Missing standard deviation values were replaced with zero.

2) Consistency Features: Daily engagement consistency was calculated by grouping interactions by student and date. For each student, the mean and standard deviation of daily interactions were computed as daily_engagement_mean and daily_engagement_std. Prior studies reported that temporal consistency in online learning behavior strongly correlates with academic achievement and dropout risk [12], [28], [29]
3) Activity Preference Features: Activity type preferences were derived using cross-tabulation of students against VLE activity types (quiz, forumng, homepage, resource, oucontent). Each cell value represents the count of interactions with that activity type. Similar activity-based behavioral modeling approaches have been adopted in adaptive learning and learning analytics research [14], [16].
4) Assessment Features: From the StudentAssessment table, the following metrics were calculated: avg_score (mean assessment score), std_score (standard deviation of scores), min_score (minimum score), max_score (maximum score), and assessments_taken (count of completed assessments). Assessment performance variables have been widely used in predictive educational analytics due to their direct relationship with student outcomes [30], [31], [32].
5) Demographic Features: Demographic variables included gender, region, highest_education, imd_band, age_band, num_of_prev_attempts, studied_credits, and disability. Categorical variables were transformed using one-hot encoding. Demographic factors were retained because previous studies reported that they may contribute to predictive performance and model stability [4], [13], [33]. However, their contribution was evaluated empirically through feature importance analysis in this study.

Regarding prediction timing, all assessment features (avg_score, std_score, assessments_taken) are based on completed assessments. In a real deployment, these features would only be available after students have submitted assessments. This study assumes that assessment data from early course modules can be used to predict final outcomes, similar to prior work [1], [11]. However, readers should note that true early

prediction would require limiting features to those available within the first few weeks of a course.

2.4 Data Preprocessing

Data preprocessing was conducted in several stages. To prevent data leakage, all preprocessing steps were applied only to the training set, with parameters (e.g., scaler means, SMOTE synthesizers) then applied to validation and test sets.

1) Target Variable Definition: The target variable was derived from the final_result column in the StudentInfo table. Following the methodology of Borna et al. [11] and Asselman et al. [34] Binary classification was applied: outcomes "Pass" and "Distinction" were coded as 1 (positive), while "Fail" and "Withdrawn" were coded as 0 (negative). The distribution was 60.4% positive (4,188 students) and 39.6% negative (2,749 students).
2) Handling Missing and Infinite Values: All feature matrices were inspected for infinite values using np.isinf(). Infinite values were replaced with NaN, then all missing values were imputed using the median of each respective column via pandas.DataFrame.fillna() [26], [27].
3) Standardization: Feature standardization was performed using sklearn.preprocessing.StandardScaler. Each feature was transformed to have zero mean and unit variance [18], [34]. Although Random Forest is a tree-based algorithm that does not require feature scaling, standardization was applied for three reasons: (a) to enable fair comparison with baseline models (e.g., Neural Network) that do require scaling; (b) to ensure consistent interpretation of standardized coefficients in extracted rules; and (c) to maintain methodological consistency with prior OULAD studies [14], [34].
4) Train-Validation-Test Split: The dataset was partitioned using a 70-15-15 split for training, validation, and testing respectively. Stratification was applied to preserve class distribution across splits [2], [35]. This split was performed before SMOTE to ensure that synthetic samples do not bleed into validation or test sets.

Table 2. Dataset Split Distribution

Split	Total Samples	Pass (Class 1)	Fail (Class 0)
Training	4855	2931	1924

Split	Total Samples	Pass (Class 1)	Fail (Class 0)
Validation	1041	629	412
Test	1041	628	413
Total	6937	4188	2749

5) Class Balancing: Class imbalance was addressed using the Synthetic Minority Over-sampling Technique (SMOTE) [31], [36]. SMOTE was applied exclusively to the training set after the train-validation-test split. The training set originally contained 4,855 samples with 1,924 minority class (Fail) samples. After SMOTE, synthetic samples were generated to balance the minority class to match the majority class (Pass).

Table 3. Training set distribution after applying SMOTE

Training Set	Pass (Class 1)	Fail (Class 0)
Before SMOTE	2931	1924
After SMOTE	2931	2931

2.5 Training Model

Random Forest was selected based on its demonstrated effectiveness in educational prediction tasks [11], [18], [30]. The algorithm operates by constructing an ensemble of decision trees during training and outputting the majority vote classification. Hyperparameters were tuned using random search with 5-fold cross-validation on the training set. The search space included: n_estimators (50, 100, 150, 200, 300), max_depth (5, 10, 15, 20, None), min_samples_split (2, 5, 10, 15, 20), min_samples_leaf (1, 2, 4, 6), and class_weight ('balanced', None). The final configuration (Table 1) was selected based on maximizing validation accuracy while preferring simpler models (smaller max_depth, higher min_samples_split) to reduce overfitting.

Table 4. Tabel Parameter Model

Parameter	Value
n_estimator	200
max_depth	15
min_samples_split	15

Parameter	Value
min_samples_leaf	2
random_state	42
max_features	sqrt
class_weight	balanced

All experiments were conducted using Python 3.10.12 with the following libraries: scikit-learn 1.3.2 (Random Forest, SMOTE, metrics), pandas 2.1.0 (data processing), numpy 1.24.3 (numerical operations), and matplotlib 3.7.2 (visualizations).

2.6 Rule Extraction

Interpretable decision rules were extracted from individual decision trees within the trained Random Forest ensemble [14], [17]. Only the first five trees from the ensemble of 200 were sampled for rule extraction. The extraction method followed the recursive tree traversal algorithm described by Schonlau [18] and further refined by Hooshyar et al. [24].

Rule extraction was performed on a subset of five decision trees rather than the entire Random Forest ensemble. The primary objective of rule extraction in this study was not to reconstruct the complete predictive behavior of the ensemble but to generate a manageable set of human-readable rules for educational interpretation. Extracting rules from all 200 trees would produce a very large number of overlapping decision paths, substantially reducing interpretability and increasing redundancy [17], [18]. Therefore, a small subset of trees was selected to balance explanatory clarity and rule diversity. Future studies may investigate systematic tree-selection strategies or ensemble-wide rule extraction methods.

For each sampled decision tree, the tree structure was traversed from root to leaf nodes using depth-limited recursion (maximum depth 2 to 3) to maintain human readability. Each extracted rule contained: (1) a conjunction of conditions (e.g., "std_clicks ≤ -0.538 AND daily_engagement_std ≤ -1.126"), (2) the predicted class, (3) the probability of failure, and (4) the number of samples reaching that leaf. Rules with fewer than 10 samples were excluded to ensure statistical reliability.

Educator validation: While the extracted rules have not been formally validated by educators in this study, the interpretability score (Section 2.8.2) and the rule format (2-3 conditions per rule) were designed to align with recommendations from prior research on teacher-facing analytics [8], [24]. Future work should include focus groups or surveys with educators to assess the practical usefulness of these rules.

2.7 Decision Engine

A decision engine was developed to map predicted risk probabilities to specific adaptive content delivery actions [5]. The engine was designed based on the pedagogical scaffolding framework proposed by Guevara-Reyes et al. [14], which categorizes intervention intensity according to risk level.

The Random Forest model outputs probability p as the probability of passing (class 1). To align with the decision engine's risk-based terminology, risk probability is defined as (1 - p), where higher risk probability indicates greater likelihood of failure. Risk thresholds were defined empirically based on the distribution of predicted probabilities in the validation set: Low Risk ($p < 0.3$), Medium Risk ($0.3 \leq p < 0.7$), and High Risk ($p \geq 0.7$) [15]. All thresholds ($p < 0.3$ for low risk, etc.) refer to this risk probability. Action Mapping: For each risk level, the engine assigned a specific set of adaptation actions, as shown in Table 5.

Table 5. Action Mapping For Recommendation

Risk Level	Recommended Action	Content Type	Scaffolding
Low ($p < 0.3$)	Enrichment	Advanced materials	Minimal
Medium ($0.3 \leq p < 0.7$)	Light scaffolding	Standard with support	Moderate
High ($p \geq 0.7$)	Full remediation	Personalized path	Maximum

While demographic features (e.g., gender, region, and disability status) were included in the predictive model because prior studies reported that they may contribute to predictive performance and model stability [4], [13], [33] their use in educational decision-making raises potential fairness concerns [37]. In the proposed framework, demographic variables are not directly used by the decision engine when assigning adaptive actions. Instead, recommendations are generated solely from the predicted failure-risk probability (p), which is derived from the complete feature set. Formal fairness metrics (e.g., demographic parity, equal opportunity, or equalized odds) were not evaluated in this

study. Therefore, no fairness-related conclusions can be drawn from the current results. Future work should investigate fairness-aware optimization, demographic subgroup analysis, and bias mitigation strategies before real-world deployment. The findings are discussed as a consideration for future fairness-aware learning analytics research. Decision Engine Pseudo-Code as shown in Algorithm 1.

```
Algorithm 1. Decision Engine for Adaptive Content Recommendation

Input:
  - P_pass (float, 0.0-1.0): predicted probability of passing
  - student_id (optional): for logging purposes
Output:
  - risk_level: "Low" | "Medium" | "High"
  - recommended_action: "Enrichment" | "Light scaffolding" | "Full
remediation"
  - content_type: string
  - scaffolding_level: "Minimal" | "Moderate" | "Maximum"

Begin
    Compute failure-risk probability:
        p = 1 - P_pass

    If p < 0.30 then
        risk_level = "Low"
        recommended_action = "Enrichment"
        content_type = "Advanced materials"
        scaffolding_level = "Minimal"

    Else if 0.30 ≤ p < 0.70 then
        risk_level = "Medium"
        recommended_action = "Light scaffolding"
        content_type = "Standard content with guided support"
        scaffolding_level = "Moderate"

    Else
        risk_level = "High"
        recommended_action = "Full remediation"
        content_type = "Personalized remedial materials"
        scaffolding_level = "Maximum"

    Return risk_level, recommended_action, content_type,
scaffolding_level
End
```

The decision engine was implemented as a Python function that accepts a risk probability and optional student context features to personalize recommendations. This function was applied to all test set predictions to generate adaptive content delivery simulations.

2.8 Evaluation

1) Standard Evaluation Metric

To evaluate the classification performance, several standard evaluation metrics were employed, including Accuracy, Precision, Recall, and F1-score [14], [31]. These metrics are widely used in classification tasks to measure prediction correctness, class-wise relevance, and balance between precision and recall [12], [18].

2) Proposed Evaluation Metric

In addition to standard classification metrics, this study proposes a custom evaluation metric[15]. Three proposed metrics were introduced to evaluate the effectiveness of the adaptive content delivery framework, addressing the absence of such metrics in prior research [5], [14].

a) Adaptation Gain (AG) was defined as the expected improvement in student success rates resulting from adaptive content delivery [5], [15]. The metric was calculated separately for high-risk and medium-risk students:

$$AG = (FRhigh \times 0.45) + (FRmedium \times 0.25) \quad (1)$$

b) where FRhigh is the actual failure rate among high-risk students and FRmedium is the actual failure rate among medium-risk students.

c) Justification for coefficients (0.45 and 0.25): These coefficients represent conservative estimates of intervention effectiveness based on prior educational intervention studies [5], [38]. Taqatqeh and Agoyi [5] reported that personalized interventions achieved a 73% passing rate compared to 45% for control groups, yielding an effective improvement of approximately 0.45 for high-risk students. The medium-risk coefficient (0.25) is a lower estimate because medium-risk students require less intensive intervention. These coefficients are assumptions for simulation purposes; actual effectiveness would need to be validated through randomized controlled trials.

Risk Coverage (RC) : was defined as the proportion of students identified as high-risk (p ≥ 0.7) in the test set:

$$\delta) \quad RC = \frac{Nhigh}{Ntotal} \quad (2)$$

where Nhigh is the number of students classified as high-risk (p ≥ 0.7), and Ntotal is the total number of students in the test set [15]. Higher Risk Coverage indicates broader identification of high-risk students, enabling more targeted interventions. [5].

Interpretability Score (IS) was calculated based on the average number of conditions per decision rule [14], [17]:

$$IS = max(0,\ min(1{,}1 - \frac{(\bar{C}-2)}{8})) \quad (3)$$

where is $\bar{C}$ denotes the average number of conditions across all extracted rules. The score ranges from 0 (low interpretability) to 1 (high interpretability) [14]. The denominator 8 represents the maximum acceptable number of conditions before a rule becomes impractical for classroom implementation [17]. When $\bar{C} \leq 2$, the score is capped at 1.000 (perfect interpretability). This metric follows the principle that rules with 2-3 conditions are readily understandable by educators, while rules with more than 10 conditions become increasingly difficult to interpret.

3) Statistical Validation

To ensure robust results, five-fold cross-validation was performed on the training set [11], [18]. Cross-validation was implemented using sklearn.model_selection.cross_val_score with the Random Forest classifier [18], [31]. The mean accuracy and standard deviation were calculated across folds. For model comparison, McNemar's test was used instead of paired t-test because both models are applied to the same test set, producing paired binary predictions (correct/incorrect). McNemar's test is specifically designed for paired nominal data and is more appropriate for comparing classification algorithms on the same dataset [12], [31]. The test was implemented using statsmodels.stats.contingency_tables.mcnemar. A significance level of α = 0.05 was adopted.

3. RESULTS AND DISCUSSION

The following section details experimental results corresponding to the three research objectives: (1) development of a predictive model for student performance, (2) extraction of interpretable decision rules for educators, and (3) creation of a decision engine that maps risk probabilities to adaptive content delivery actions.

3.1. Predictive Model Performance

The Random Forest model was trained using 4,855 samples and 27 features. Table 6 summarizes the model's performance on the test set, which included 1,041 samples.

Table 6. Random Forest Performance Metric

Evaluation Metric	Value
Accuracy	0.8722
Precision	0.8689
Recall	0.9283
F1-Score	0.8976
AUC-ROC	0.9324

The model achieved an accuracy of 87.22% and an AUC-ROC of 0.9324, demonstrating strong discriminative capability between passing and failing students. Five-fold cross-validation further confirmed model stability, yielding a mean accuracy of 88,98% with standard deviation of ±0.0468

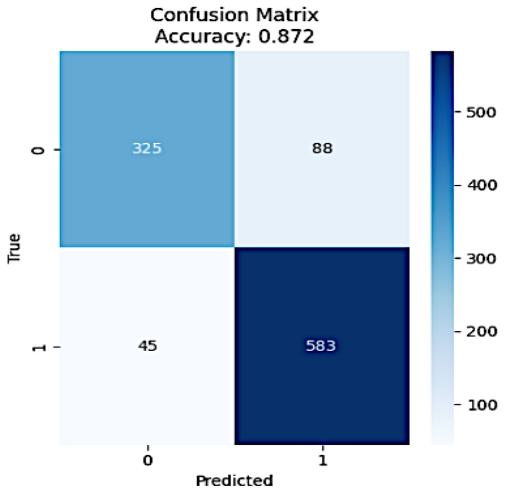

Figure 2. Confusion Matrix

Table 7. Confusion Matrix (Test Set, n=1,041)

	Predicted Pass	**Predicted Fail**
Actual Pass	325 (True Positive)	88 (False Negative)
Actual Fail	45 (False Positive)	583 (True Negative)

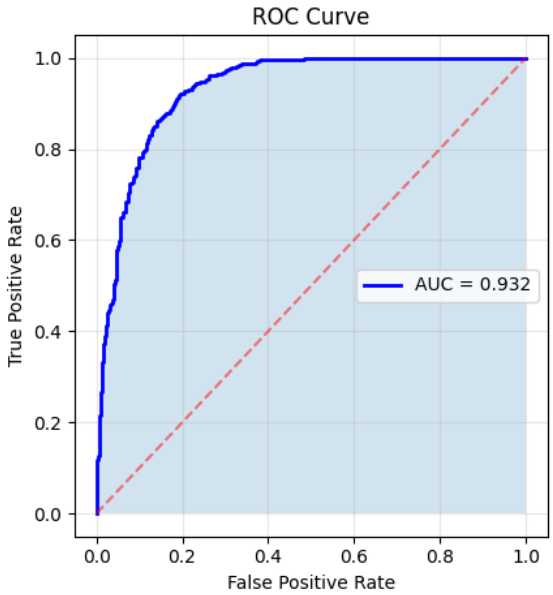

Figure 3. ROC Curve (AUC = 0.9324)

From the Table.4 implies that the model correctly identified 325 passing students and 583 failing students, with 45 false negatives (failing students incorrectly predicted as passing) and 88 false positives (passing students flagged as at-risk). The ROC curve demonstrates strong discriminative capability of the proposed Random Forest model, with an AUC value of 0.9324 indicating excellent separation between pass and fail students across multiple probability thresholds. To evaluate the contribution of each component and compare against other models, an ablation study and baseline comparison were conducted:

Table 8. Baseline Model Comparison

Model	**Accuracy**	**Precision**	**Recall**	**F1 Score**	**AUC**
RF (Proposed)	0.8722	0.8689	0.9283	0.8976	0.9324
Logistic Regression	0.8377	0.8756	0.8519	0.8636	0.8926
XGBoost	0.8866	0.8960	0.9188	0.9072	0.9350
Neural Network (MLP)	0.8405	0.8850	0.8455	0.8648	0.9045

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

Competitive performance with 87.22% accuracy, 89.76% F1-score, and 0.9324 AUC, while also obtaining the highest recall value (92.83%). Because the positive class was defined as Pass (Class 1), the reported Recall metric represents the proportion of passing students correctly identified by the model. Although XGBoost slightly outperformed the proposed model in overall predictive performance, the Random Forest framework provides a better balance between predictive capability and interpretability through its rule extraction and adaptive recommendation components. In contrast, Logistic Regression and Neural Network models showed lower overall performance, particularly in recall and AUC metrics. McNemar tests were conducted to compare the proposed Random Forest framework against baseline models using the same test set (1,041 samples).

Table 9. Statistical Significance Tests

Comparison	Both Correct	A only correct	B only correct	Both Wrong	χ^2	p-value	Significant
Random Forest (Proposed) vs Logistic Regression	838	70	34	99	11.7788	<0.001	YES
Random Forest (Proposed) vs XGBoost	888	20	35	98	3.5636	0.0591	NO
Random Forest (Proposed) vs Neural Network (MLP)	837	71	38	95	9.3945	0.0022	YES

McNemar's test indicated statistically significant differences between the proposed Random Forest model and Logistic Regression (χ^2 = 11.78, $p < 0.001$), as well as between Random Forest and MLP (χ^2 = 9.39, $p = 0.002$). However, the difference between Random Forest and XGBoost was not statistically significant (χ^2 = 3.56, $p = 0.059$), suggesting comparable predictive performance between the two models. Therefore, Random Forest was selected not because it achieved the highest predictive accuracy, but because it provides a more favorable balance between predictive performance, interpretability, and direct rule-extraction capability.

3.2. Validation Set Performance and Threshold Determination

Risk thresholds for the decision engine were determined empirically based on the validation set (1,041 samples). The validation set performance is reported in Table 10. Analysis of different thresholds revealed. The optimal threshold for maximizing F1-score was 0.48 (F1=0.8957). However, threshold 0.7 was selected for the decision engine because it provides the highest precision (92.5%), minimizing false positives and ensuring that flagged high-risk students genuinely need intervention.

Table 10. Threshold Analysis on Validation Set

Threshold	Precision	Recall	High-Risk (%)
0.3	0.8187	0.9762	72.0%
0.5	0.8694	0.9205	64.0%
0.7	0.9250	0.8045	52.5%

3.3. Interpretable Decision Rules

Decision rules were extracted from the Random Forests ensemble using a recursive tree traversal algorithm. Rules were extracted from the first five trees of the ensemble (sampled from 200 total trees). From these five trees, a total of 20 raw rules were extracted. After removing duplicate rules (rules with identical conditions) and applying a minimum sample threshold of 10 samples per rule, 20 unique rules remained.

Table 11. Rule Extraction Quality Metrics

Metric	Value
Total Tree sampled	5 (of 200)
Raw rules extracted	20
Unique rules (after deduplication)	20
Fidelity (agreement with full RF)	81.6%
Coverage (test set)	100%
Average confidence	0.777
Average training support	927.8 samples per rule
Average conditions per rule	2.0
Interpretability Score	1.000

Fidelity of 81.6% indicates substantial agreement between the extracted rule set and the full Random Forest model. The coverage of 100% confirms that all test samples are covered by at least one rule. Average training support of 927.8 samples per rule refers to the number of training samples reaching the corresponding leaf node during rule extraction. The interpretability score of 1.000 (perfect) reflects that all rules have ≤2 conditions, making them readily understandable by educators without technical expertise.

Table 12. Sample Decision Rules Extracted From Random Forest

Rules	Condition	Prediction	Failure Probability	Risk Level
1	IF std_clicks ≤ -0.550 AND daily_engagement_std ≤ -1.006	Fail	0.92	High
2	IF std_clicks ≤ -0.550 AND daily_engagement_std > -1.006	Fail	0.70	High
3	IF std_clicks > -0.550 AND assessments_taken ≤ -0.585	Fail	0.98	High
4	IF std_clicks > -0.550 AND assessments_taken > -0.585	Pass	0.27	Low
5	IF pref_quiz ≤ -0.076 AND unique_days ≤ -0.519	Fail	0.87	High
6	IF pref_quiz ≤ -0.076 AND unique_days > -0.519	Pass	0.34	Medium
7	IF pref_quiz > -0.076 AND unique_days ≤ -0.336	Fail	0.54	Medium
8	IF pref_quiz > -0.076 AND unique_days > -0.336	Pass	0.20	Low

3.4. Feature Importance Analysis by Category

Feature importance was analyzed by grouping features into five categories, as shown in Table 13. Assessment features dominate (41.9%), with assessments_taken as the single most important feature (27.12%). This indicates that consistent engagement with evaluative activities is the strongest predictor of student success. Engagement features (29.4%) are the second most important category, followed by Activity Preference

features (22.5%). Demographic features have a minimal contribution (1.3%), suggesting that behavioral factors are far more predictive than static demographic characteristics.

Table 13. Feature Importance by Category

Category	Total Importance	% of Total	Features Count
Assessment	0.4189	41.9%	5
Engagement	0.2939	29.4%	5
Consistency	0.0495	5.0%	2
Activity Preference	0.2245	22.5%	13
Demographic	0.0132	1.3%	2

3.5. Decision Engine for Adaptive Content Delivery

The decision engine assigned predicted risk probabilities to recommendations according to three defined risk tiers (thresholds determined from validation set: $p < 0.3$ for low risk, $0.3 \leq p < 0.7$ for medium risk, $p \geq 0.7$ for high risk). Table 14 displays the risk distribution among the 1,041 test students.

Table 14. Risk Distribution And Recommended Action

Risk Level	Threshold	Number of students	Actual Fail	Fail Rates	Population	Recommended Action
Low	$p < 0.3$	533	42	0.079	51.2%	Enrichment (challenge-based learning)
Medium	$0.3 \leq p < 0.7$	214	90	0.421	20.6%	Light scaffolding (guided learning)
High	$p \geq 0.7$	294	281	0.956	28.2%	Full remediation (personalized intervention)

The decision engine classified 533 students (51.2%) as low risk, 214 students (20.6%) as medium risk, and 294 students (28.2%) as high risk. The actual failure rates increased consistently across risk levels, from 7.9% in the low-risk group to 42.1% in the medium-risk group and 95.6% in the high-risk group. This pattern indicates that the risk probability

thresholding was consistent with actual student outcomes. The Adaptation Gain is then computed as:

$$AG = (FRhigh \times 0.45) + (FRmedium \times 0.25)$$
$$AG = (0.956 \times 0.45) + (0.421 \times 0.25)$$
$$AG = 0.4302 + 0.1052$$
$$AG = 0.5354$$

Detailed Example of Adaptive Content Recommendation as Table 15.

Table 15. Detailed Adaptive Content Recommendation Example (Student ID: 10)

Component	Details
Student Profile	Risk Score : 0.785; Actual outcome: Fail;
Risk Factors	Low quiz engagement (pref_quiz = -0.132), below-average unique days, high score variability
Recommended Action	Full remediation with personalized path
Content Type	Remedial/Alternative materials with fundamental concept review
Recommended Resources	One-on-one tutoring, prerequisite concept review, concept mapping, guided practice
Expected Success Probability	0.65
Pedagogical Rationale	Student shows inconsistent engagement and performance variability, suggesting gaps in foundational knowledge

3.6. Proposed Metrics

Table 16 presents the novelty metrics introduced in this study. Adaptation Gain: The total Adaptation Gain was 0.5354, indicating that the adaptive content delivery system is projected to improve student success rates by 53.54% compared to no intervention. This value was derived from the expected improvement for high-risk students (0.4302) combined with that for medium-risk students (0.1052). The Adaptation Gain represents an estimated improvement based on conservative assumptions (45% improvement for high-risk students, 25% for medium-risk students) derived from prior intervention studies [5], [38]. Actual effectiveness would require validation through randomized controlled trials with real intervention deployment. Risk Coverage: The model identified 28.2% of

students as high-risk (p ≥ 0.7), enabling targeted intervention for 294 students. At this threshold, precision reached 92.6%, indicating that 92% of flagged students genuinely required intervention. Interpretability Score: With an average of 2.0 conditions per rule, the interpretability score reached 1.000, confirming that the extracted rules are optimally understandable for classroom implementation. Twenty unique rules were extracted in total. The average number of conditions per rule was 2.0, and the interpretability score reached 1.000, indicating that all rules are readily understandable by educators without technical expertise.

Table 16. Proposed Metric Result

Metric	Value	Interpretation
Adaptation Gain (estimated)	53.54%	Expected 53.54% improvement in success rates
Risk Coverage	28.2% (294 students)	High-risk students identified for intervention
Interpretability Score	1.000	Perfect interpretability (all rules ≤2 conditions)
Precision at p≥0.7	92.1%	High precision for high-risk classification

3.7. Discussion

This study proposed a Random Forest-based framework to bridge the prediction-to-action gap in student performance prediction [5], [14]. The following discussion addresses the research questions, interprets key findings, acknowledges limitations, and provides practical implications.

3.7.1. Prediction Performance (RQ1)

The proposed Random Forest model achieved 87.22% accuracy and 0.9324 AUC-ROC on the test set, with a cross-validation mean accuracy of 0.8898 (±0.0468). These results answer RQ1 by demonstrating that Random Forest can effectively predict student performance (pass/fail) using OULAD data. Compared to prior studies on the same dataset, our model shows competitive performance. Borna et al. [11] reported 78.68% accuracy using Random Forest for withdrawal prediction, while Torkhani and Rezgui [25]

found that LSTM networks achieved 83.41% accuracy. Our framework achieves 87.22% accuracy, outperforming these benchmarks. The high recall (92.83%) is particularly significant for educational applications, as false negatives (failing students incorrectly predicted as passing) would not receive necessary interventions [4], [11].

The five-fold cross-validation mean accuracy of 88.98% with a low standard deviation (±4.68%) confirms the stability and robustness of the model. The minimal gap between training accuracy (99.87%) and test accuracy (87.22%) suggests that while the model fits the training data well, there is room for improvement in generalization.

It is important to acknowledge that XGBoost achieved marginally higher predictive accuracy (88.66% vs. 87.22%) and AUC (0.9350 vs. 0.9324) compared to the proposed Random Forest model. However, McNemar's test indicated that this difference was not statistically significant ($p = 0.059$). Therefore, Random Forest was selected not because it outperforms XGBoost in predictive accuracy, but because it offers a more favorable trade-off between predictive performance, inherent interpretability (via tree structures), and direct rule-extraction capability without requiring post-hoc explanation methods like SHAP or LIME. For educational applications where explainability is as important as accuracy, this trade-off is justified. In summary, model selection in this study prioritized interpretability, rule extraction capability, and actionability rather than marginal predictive gains.

The number of assessments completed emerged as the strongest predictor (27.12% importance). This finding has important practical implications: consistent engagement with evaluative activities is more critical than raw click counts or passive content consumption [1]. However, this also introduces an early-warning timing issue: assessment data may only become available after students have completed assignments, potentially delaying intervention. In practice, early indicators (e.g., unique_days, interaction_count in the first two weeks) could provide earlier warnings, though with potentially lower accuracy.

3.7.2. Interpretable Decision Rules (RQ2)

With an average of 2.0 conditions per rule and a perfect interpretability score (1.000), this study demonstrates that complex machine learning models can produce explanations

understandable by educators without technical expertise [14], [17]. The rule fidelity of 81.6% and coverage of 100% confirm that the extracted rule set closely approximates the full Random Forest model's decisions while remaining interpretable.

3.7.3. Comparison with Decision Rules (RQ3 & RQ4)

Unlike existing early warning systems [6], [7] that stop at risk identification, or recommendation systems [5] that suggest interventions without automated content mapping, this framework provides an end-to-end pipeline from prediction to action. The decision engine produces simulated recommendations that educators can use to guide intervention strategies. Table 17 summarizes the comparison.

Table 17. Comparison with Prior Systems

Feature	Akçapınar et al. [6]	Taqatqeh and Agoyi [5]	Guevara-Reyes et al. [14]	This Study
Prediction model	Logistic Regression	XGBoost	Randon Forest	Random Forest
Interpretability	Low	Low (SHAP)	Medium (SHAP)	High (rule extraction)
Early warning	✓	✓	×	✓
Recommendation engine	×	✓ (text recommendation)	×	✓ (decision engine)
Automated content mapping	×	×	×	✓
Adaptation Gain metric	×	×	×	✓
Rule fidelity reported	×	×	×	✓ (81.6%)

The framework identified 28.2% of students (294 out of 1,041) as high-risk. While this enables broad coverage, it may create a substantial intervention workload for educators. To address this, institutions should implement a tiered prioritization strategy:

1) Priority 1 (Critical): Students with $p \geq 0.9$ (approximately 15-20% of high-risk students) - require immediate one-on-one intervention
2) Priority 2 (High): Students with $0.8 \leq p < 0.9$ (approximately 30-40% of high-risk students) - require weekly check-ins

3) Priority 3 (Moderate): Students with $0.7 \leq p < 0.8$ (approximately 40-55% of high-risk students) - automated content adaptation with periodic monitoring

Practical LMS deployment would require additional infrastructure to support real-time analytics, intervention delivery, and feedback-driven model improvement, such as [24]:

1) LMS plugin to extract real-time student interaction data
2) Teacher dashboard displaying risk classifications and recommended actions for each student
3) Intervention tracking system to log which interventions were delivered
4) Feedback loop to capture intervention outcomes and refine the model over time
5) Use of Demographic Features

The predictive model includes demographic variables (gender, region, highest_education, imd_band, age_band, disability, studied_credits, and previous attempts) because prior educational data mining studies have reported that such variables may improve predictive performance and model stability [4], [13], [33]. However, feature importance analysis showed that demographic variables contributed only 1.3% of the total predictive importance, substantially lower than assessment-related features (41.9%) and engagement-related features (29.4%). This finding suggests that behavioral and academic activity features are considerably more influential than demographic characteristics in predicting student outcomes.

The decision engine does not directly use demographic variables when assigning adaptive actions. Instead, recommendations are generated solely from the predicted failure-risk probability. Nevertheless, because demographic variables are included in the predictive model, potential fairness concerns remain. Formal fairness metrics and demographic subgroup analyses were not conducted in this study; therefore, fairness-related conclusions cannot be drawn. Future work should evaluate demographic fairness using established metrics and investigate bias mitigation strategies prior to operational deployment. As emphasized by Leppan et al.[37], ethical learning analytics frameworks should ensure transparency and fairness in algorithmic decision-making.

Labeling students as "high-risk" carries potential negative consequences, including student demotivation, self-fulfilling prophecies, and privacy concerns [8]. To address these concerns:

1) Risk labels should be educator-facing only, not visible to students
2) Intervention content should be framed as "additional support" rather than "remediation"
3) Students should have the option to opt out of automated interventions
4) Regular human review of automated recommendations should be conducted

Several limitations of this study should be acknowledged:

1) Dataset and Generalizability: While the OULAD dataset is comprehensive, findings may not generalize to all educational contexts, as Kuzilek et al. [1] note that the Open University's distance learning model differs from traditional residential universities [3]. The selection of only two course modules (AAA and BBB) may further limit generalizability to smaller courses or courses with different assessment structures.
2) Prediction Timing and Early Warning: Assessment features (avg_score, std_score, assessments_taken) are only available after students complete assessments. In a true early warning system deployed early in a course, these features would not yet be available. The reported accuracy (87.22%) reflects end-of-course prediction rather than early prediction. Future work should evaluate performance using only features available within the first 2-4 weeks.
3) Binary Classification Limitation: The study addresses binary prediction (pass/fail) rather than fine-grained grade prediction. Rimal and Sharma [13] demonstrated that multiclass grade prediction remains challenging, with only 66% accuracy for A-grade prediction.
4) SMOTE and Synthetic Data: SMOTE was applied to address class imbalance, generating synthetic minority class samples. While this improved model performance, synthetic samples may not perfectly represent real student patterns, and results may not fully generalize to the original imbalanced distribution.
5) Lack of Temporal Validation: The current approach uses random splitting, which may overestimate performance because interactions from later course presentations are mixed with earlier ones. A stricter temporal validation would

train models on earlier presentations (e.g., 2013J) and test on later presentations (e.g., 2014J). This is acknowledged as a limitation and recommended for future work.

6) Rule Extraction Sampling: Decision rules were extracted from the first five trees of the 200-tree Random Forest ensemble. Although rule fidelity (81.6%) and coverage (100%) indicate that the extracted rules are representative, using all trees could provide a more comprehensive rule set at the cost of increased redundancy. In addition, the extracted rules have not been validated by educators. While the interpretability score (1.000) suggests structural simplicity, their practical usefulness in educational settings remains unverified. Therefore, educator validation is required before real-world deployment.
7) Adaptation Gain as Estimate: The Adaptation Gain metric represents estimated improvement based on assumptions from prior studies, not measured outcomes from actual adaptive content delivery. Real-world deployment studies with randomized controlled trials are required to validate these estimates [5], [15], [38].
8) No Real Intervention Testing: The decision engine produces simulated recommendations only; no actual interventions were deployed to students, and no learning outcome improvements were measured. The framework should be validated through a full-cycle deployment study.
9) Educator Validation Not Conducted: While the interpretability score (1.000) indicates rule simplicity, the extracted rules have not been formally validated by educators. Future work should include focus groups or surveys with educators to assess practical usefulness.

Based on the limitations identified, future research should:

1) Conduct randomized controlled trials to validate the Adaptation Gain estimate with measured learning outcomes
2) Implement strict temporal validation training on earlier presentations (2013J) and testing on later presentations (2014J)
3) Extend to multiclass grade prediction (A, B, C, D, Fail) to provide finer-grained student profiling
4) Integrate additional data modalities including video engagement metrics, clickstream sequences, and student self-report measures

5) Deploy the framework in a real LMS environment with teacher dashboards and intervention tracking
6) Conduct formal fairness evaluations using demographic subgroup analysis and established fairness metrics (e.g., demographic parity, equal opportunity, and equalized odds), followed by bias mitigation strategies where necessary
7) Validate extracted rules with educator focus groups to ensure practical usability

4. CONCLUSION

This study addressed the prediction-to-action gap in student performance prediction by proposing an integrated Random Forest-based framework that links prediction, interpretability, and adaptive content recommendations. Using the Open University Learning Analytics Dataset (OULAD) filtered to 6,937 students across two course modules (AAA and BBB), the framework achieved three primary contributions. First, the predictive model attained 87.22% accuracy and 0.9324 AUC-ROC, with a cross-validation mean of 88.98%, demonstrating strong predictive capability for identifying at-risk students. Second, extraction of interpretable decision rules yielded 20 human-readable rules with an average of 2.0 conditions per rule, a perfect interpretability score of 1.000, a fidelity of 81.6% with the full Random Forest model, and an average confidence of 0.777. Third, the decision engine mapped risk probabilities to simulated adaptive content recommendations, identifying 28.2% of test students as high-risk with 92.5% precision at threshold 0.7. The proposed Adaptation Gain metric a simulation-based estimate derived from assumed intervention effectiveness coefficients (0.45 for high-risk, 0.25 for medium-risk) suggested a potential 53.54% improvement in student success rates. Importantly, this is an estimate only, not a measured outcome from actual adaptive content delivery. The decision engine was evaluated computationally only; no real interventions were deployed to students, and no learning outcome improvements were measured. The framework is presented as a simulation-based decision-support model for educators, not as a validated adaptive learning system. Several critical steps remain before practical adoption: (1) Educator validation of the extracted rules has not been conducted and is required before practical adoption; (2) Strict temporal validation training on earlier course presentations (e.g., 2013J) and testing on later presentations (e.g., 2014J) is necessary to assess generalizability across academic sessions. The current

random split may overestimate performance.; (3) real deployment in a live LMS environment with teacher dashboards and intervention tracking must be conducted; and (4) randomized controlled trials are required to measure actual intervention effectiveness. The framework's novelty lies in its end-to-end integration of high-accuracy prediction, interpretable rule extraction, and a decision engine for adaptive recommendations, offering a theoretical pathway not yet a proven solution to support student outcomes through data-driven interventions.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to the Faculty of Engineering, Universitas Negeri Surabaya, for providing the academic support and research facilities that made this study possible.

REFERENCES

[1] J. Kuzilek, M. Hlosta, and Z. Zdrahal, "Open University Learning Analytics dataset," *Sci. Data*, vol. 4, no. 1, p. 170171, Nov. 2017, doi: 10.1038/sdata.2017.171.

[2] M. Hernández-de-Menéndez, R. Morales-Menendez, C. A. Escobar, and R. A. Ramírez Mendoza, "Learning analytics: state of the art," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 16, no. 3, pp. 1209–1230, Sep. 2022, doi: 10.1007/s12008-022-00930-0.

[3] R. Conijn, C. Snijders, A. Kleingeld, and U. Matzat, "Predicting Student Performance from LMS Data: A Comparison of 17 Blended Courses Using Moodle LMS," *IEEE Transactions on Learning Technologies*, vol. 10, no. 1, pp. 17–29, Jan. 2017, doi: 10.1109/TLT.2016.2616312.

[4] K. T. Chui, D. C. L. Fung, M. D. Lytras, and T. M. Lam, "Predicting at-risk university students in a virtual learning environment via a machine learning algorithm," *Comput. Human Behav.*, vol. 107, p. 105584, Jun. 2020, doi: 10.1016/j.chb.2018.06.032.

[5] S. Taqatqeh and M. Agoyi, "Early warning and recommendation system for undergraduate students at higher education institutions based on machine learning methods," *PeerJ Comput. Sci.*, vol. 12, p. e3792, May 2026, doi: 10.7717/peerj-cs.3792.

[6] G. Akçapınar, A. Altun, and P. Aşkar, "Using learning analytics to develop early-warning system for at-risk students," *International Journal of Educational Technology in Higher Education*, vol. 16, no. 1, p. 40, Dec. 2019, doi: 10.1186/s41239-019-0172-z.

[7] G. Akçapınar, M. N. Hasnine, R. Majumdar, B. Flanagan, and H. Ogata, "Developing an early-warning system for spotting at-risk students by using eBook interaction logs," *Smart Learning Environments*, vol. 6, no. 1, p. 4, Dec. 2019, doi: 10.1186/s40561-019-0083-4.

[8] B. Francisco, O. Jeroen, and R. Lezel, "Explainable AI in education: Fostering human oversight and shared responsibility," 2025. [Online]. Available: http://europa.eu

[9] D. Gašević, S. Dawson, T. Rogers, and D. Gasevic, "Learning analytics should not promote one size fits all: The effects of instructional conditions in predicting academic success," *Internet High. Educ.*, vol. 28, pp. 68–84, Jan. 2016, doi: 10.1016/j.iheduc.2015.10.002.

[10] H. A. Alhakbani and F. M. Alnassar, "Open Learning Analytics: A Systematic Review of Benchmark Studies using Open University Learning Analytics Dataset (OULAD)," in *2022 7th International Conference on Machine Learning Technologies (ICMLT)*, New York, NY, USA: ACM, Mar. 2022, pp. 81–86. doi: 10.1145/3529399.3529413.

[11] M.-R. Borna, H. Saadat, A. T. Hojjati, and E. Akbari, "Analyzing click data with AI: implications for student performance prediction and learning assessment," *Front. Educ. (Lausanne)*, vol. 9, Dec. 2024, doi: 10.3389/feduc.2024.1421479.

[12] A. Omarbekova *et al.*, "A temporal attention-based hybrid deep learning model for student performance and academic risk prediction," *Front. Artif. Intell.*, vol. 9, May 2026, doi: 10.3389/frai.2026.1811886.

[13] Y. Rimal and N. Sharma, "Ensemble machine learning prediction accuracy: local vs. global precision and recall for multiclass grade performance of engineering students," *Front. Educ. (Lausanne)*, vol. 10, Apr. 2025, doi: 10.3389/feduc.2025.1571133.

[14] R. Guevara-Reyes, I. Ortiz-Garcés, R. Andrade, F. Cox-Riquetti, and W. Villegas-Ch, "Machine learning models for academic performance prediction: interpretability and application in educational decision-making," *Front. Educ. (Lausanne)*, vol. 10, Aug. 2025, doi: 10.3389/feduc.2025.1632315.

[15] H. Abed, "Adaptive Learning with Attention-Based Knowledge Tracing and Risk Prediction for Improved Student Outcomes," *Iraqi Journal for Applied Sciences*, vol. 3, no. 1, pp. 74–83, Mar. 2026, doi: 10.69923/mewnft50.

[16] S. K. Banihashem, M. Farrokhnia, M. Badali, and O. Noroozi, "The impacts of constructivist learning design and learning analytics on students' engagement and self-regulation," *Innovations in Education and Teaching International*, vol. 59, no. 4, pp. 442–452, Jul. 2022, doi: 10.1080/14703297.2021.1890634.

[17] D. Hooshyar and Y. Yang, "Problems With SHAP and LIME in Interpretable AI for Education: A Comparative Study of Post-Hoc Explanations and Neural-Symbolic Rule Extraction," *IEEE Access*, vol. 12, pp. 137472–137490, 2024, doi: 10.1109/ACCESS.2024.3463948.

[18] M. Schonlau and R. Y. Zou, "The random forest algorithm for statistical learning," *The Stata Journal: Promoting communications on statistics and Stata*, vol. 20, no. 1, pp. 3–29, Mar. 2020, doi: 10.1177/1536867X20909688.

[19] L. S. VYGOTSKY, *Development of Higher Psychological Processes*. Harvard University Press, 1978. doi: 10.2307/j.ctvjf9vz4.

[20] D. Wood, J. S. Bruner, and G. Ross, "THE ROLE OF TUTORING IN PROBLEM SOLVING," *Journal of Child Psychology and Psychiatry*, vol. 17, no. 2, pp. 89–100, Apr. 1976, doi: 10.1111/j.1469-7610.1976.tb00381.x.

[21] C. A. Tomlinson, *The Differentiated Classroom: Responding to the Needs of All Learners*, 2nd ed. ASCD, 2014.

[22] J. VanTassel-Baska, "Are We Differentiating Effectively for the Gifted or Not? A Commentary on Differentiated Curriculum Use in Schools," *Gifted Child Today*, vol. 42, pp. 165–167, Jul. 2019, doi: 10.1177/1076217519842626.

[23] B. Belland, "Instructional Scaffolding: Foundations and Evolving Definition," 2017, pp. 17–53. doi: 10.1007/978-3-319-02565-0_2.

[24] P. Brusilovsky, "Adaptive Hypermedia for Education and Training," in *Adaptive Technologies for Training and Education*, Cambridge University Press, 2012, pp. 46–66. doi: 10.1017/CBO9781139049580.006.

[25] W. Torkhani and K. Rezgui, "OULAD MOOC Student Performance Prediction using Machine and Deep Learning Techniques," in *Proceedings of the International Conference on Decision Aid and Artificial Intelligence (ICODAI 2024)*, Atlantis Press, 2025, pp. 228–241. doi: 10.2991/978-94-6463-654-3_18.

[26] B. Albreiki, N. Zaki, and H. Alashwal, "A Systematic Literature Review of Student' Performance Prediction Using Machine Learning Techniques," *Educ. Sci. (Basel).*, vol. 11, no. 9, p. 552, Sep. 2021, doi: 10.3390/educsci11090552.

[27] A. N. Firdaus and Y. R. Sipayung, "Stacking Ensemble Learning for University Student Dropout Prediction," *Journal of Information Systems and Informatics*, vol. 8, no. 1, pp. 456–477, Feb. 2026, doi: 10.63158/journalisi.v8i1.1403.

[28] Y. He *et al.*, "Online At-Risk Student Identification using RNN-GRU Joint Neural Networks," *Information*, vol. 11, no. 10, p. 474, Oct. 2020, doi: 10.3390/info11100474.

[29] U. Biswas, S. Sinha, and P. De, "Recurrent Neural Network Model to Predict and Monitor Student Performance," in *2025 8th International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech)*, IEEE, Jan. 2025, pp. 1–6. doi: 10.1109/IEMENTech65115.2025.10959550.

[30] S. Dass, K. Gary, and J. Cunningham, "Predicting Student Dropout in Self-Paced MOOC Course Using Random Forest Model," *Information*, vol. 12, no. 11, p. 476, Nov. 2021, doi: 10.3390/info12110476.

[31] N. A. Butt, Z. Mahmood, K. Shakeel, S. Alfarhood, M. Safran, and I. Ashraf, "Performance Prediction of Students in Higher Education Using Multi-Model Ensemble Approach," *IEEE Access*, vol. 11, pp. 136091–136108, 2023, doi: 10.1109/ACCESS.2023.3336987.

[32] D. Dhiyaussalam, A. Yusuf, I. Wardiah, and N. L. Putri, "Predicting Respiratory Conditions Using Random Forest and XGBoost," *Journal of Information Systems and Informatics*, vol. 7, no. 2, pp. 1642–1657, Jun. 2025, doi: 10.51519/journalisi.v7i2.1124.

[33] F. T. Johora, M. N. Hasan, A. Rajbongshi, M. Ashrafuzzaman, and F. Akter, "An explainable AI-based approach for predicting undergraduate students academic performance," *Array*, vol. 26, p. 100384, Jul. 2025, doi: 10.1016/j.array.2025.100384.

[34] A. Asselman, M. Khaldi, and S. Aammou, "Enhancing the prediction of student performance based on the machine learning XGBoost algorithm," *Interactive Learning Environments*, vol. 31, no. 6, pp. 3360–3379, Aug. 2023, doi: 10.1080/10494820.2021.1928235.

[35] L. S. Riza, L. A. Bexheti, and J. Zoroja, "Early Identification of At-Risk Students in Online Education: A Deep Learning Approach to Predictive Modelling," *Business Systems Research Journal*, vol. 16, no. 2, pp. 69–91, Dec. 2025, doi: 10.2478/bsrj-2025-0019.

[36] A. Fernandez, S. Garcia, F. Herrera, and N. V. Chawla, "SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary,"

Journal of Artificial Intelligence Research, vol. 61, pp. 863–905, Apr. 2018, doi: 10.1613/jair.1.11192.

[37] R. G. Leppan, R. A. Botha, and J. F. Van Niekerk, "Process Model for Differentiated Instruction using Learning Analytics," *South African Computer Journal*, vol. 30, no. 2, Dec. 2018, doi: 10.18489/sacj.v30i2.481.

[38] A. Farida, V. Atina, and D. Suwandi, "Mathematical Modeling and Integration of Machine Learning-Based Prediction System on E-Learning Platform to Improve Students' Academic Performance," *JTAM (Jurnal Teori dan Aplikasi Matematika)*, vol. 9, no. 3, p. 829, Jul. 2025, doi: 10.31764/jtam.v9i3.30994.