JRMM – Jurnal Riset Mahasiswa Matematika, Volume 5 (4) (2026), Pages 267-277

Research Article

Deep Neural Network-Based Student Performance Prediction with Hessian-Free Optimization

Andy Irawan[1], Zainal Abidin[1], and Mohammad Jamhuri[2]

[1] *Department of Informatics Engineering, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia*

[2] *Department of Mathematics, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim, Malang, Indonesia*

Article History

Received December 02, 2025
Revised February 03, 2026
Accepted March 30, 2026
Published April 30, 2026

Abstract. Predicting student graduation predicates is important for academic monitoring and timely intervention in higher education. This study investigates graduation predicate prediction using deep neural networks under three feature-group settings: academic-only, non-academic-only, and combined academic–non-academic features. A multilayer perceptron with three hidden layers was trained using SGD with momentum, RMSProp, Adam, and a damped Hessian-free optimization procedure. Two tasks were considered: a four-class graduation predicate classification task and a binary risk-screening task in which *Sufficient* was treated as the positive risk class. The results show that the combined feature group achieved the best multiclass performance, with an accuracy of 0.8478 and a weighted F1-score of 0.8274. Hessian-free optimization consistently produced the best results across all feature-group scenarios, with the clearest gain appearing in the non-academic-only setting. In the additional risk-screening analysis, non-academic variables provided meaningful but limited predictive signal, and *Major* emerged as the strongest individual predictor. These findings show that combining academic and non-academic information improves graduation predicate prediction and that Hessian-free optimization is an effective training strategy for deep neural classification in educational data.

Keywords: deep neural networks; educational data mining; graduation predicate prediction; Hessian-free optimization; risk screening.

1. Introduction

Predicting student outcomes has become an important topic in educational data mining and learning analytics because it can support early intervention, targeted academic assistance, and institutional decision-making [1–3]. Within this broader area, graduation predicate prediction is especially relevant because it condenses cumulative academic performance into an interpretable final achievement category that is meaningful for students, study programmes, and institutions [4, 5].

A large body of previous work has relied primarily on academic variables, including prior grades, semester GPA, course performance, and learning-management-system activity. These variables often provide strong predictive signal because they directly reflect student progress. Several studies have shown that early-semester academic performance is among the strongest predictors of final academic outcomes [6, 7]. Their practical limitation, however, is temporal: they become informative only after students have already progressed through part of their studies. This reduces their value for admission-time screening and very early intervention, when support may be most useful.

For that reason, non-academic and admission-related variables remain important, particularly when academic records are incomplete or unavailable. Prior studies have reported that factors such as gender, school background, admission pathway, organizational involvement, and demographic characteristics may contribute to the prediction of student outcomes, although usually less strongly than academic indicators [8–10]. Admission-based prediction has likewise been shown to be feasible for anticipating later academic performance [11–13]. In the context of Islamic higher education, this issue is especially interesting because variables such as boarding-school experience and Arabic-language proficiency may capture forms of prior preparation that are rarely examined in the broader educational data mining literature. These variables may not dominate prediction on their own, but they may still provide useful contextual information, especially in early-stage settings.

From the modelling perspective, educational data mining studies have employed a broad range of machine learning methods, including Naïve Bayes, Decision Tree, K-Nearest Neighbors, Support Vector Machine, and neural-network-based models [7, 14–17]. These studies demonstrate that educational data can support useful prediction, yet most of them focus on classifier comparison, feature selection, or predictive accuracy. Comparatively less attention has been given to the optimization procedure used to train neural models, even though the optimizer can materially affect convergence behaviour, stability, and final predictive performance. This issue becomes particularly relevant when the input space combines continuous academic variables with high-dimensional encoded categorical features, because the resulting loss surface may be difficult for purely gradient-based training.

In deep learning, first-order methods such as stochastic gradient descent, RMSProp, and Adam are widely used because of their simplicity and computational efficiency. Neverthe-

Corresponding author's email: m.jamhuri@live.com
DOI: https://doi.org/10.18860/JRMM.v5i4.37951 / p-ISSN: 2086-0382 | e-ISSN: 2477-3344 (BRIN | ISSN Portal)

less, second-order methods remain attractive because they exploit curvature information of the objective function and can therefore yield more informative search directions than gradient information alone. Among these methods, Hessian-free optimization is especially appealing because it avoids explicit construction of the Hessian matrix and instead relies on matrix–vector products computed efficiently by automatic differentiation [18, 19]. The resulting inner linear system can then be solved iteratively by the conjugate gradient method, making second-order training practical for neural networks [20]. Although current neural-network optimization research is still dominated by first-order methods, curvature-aware and second-order approaches continue to attract attention because they may provide stronger convergence behaviour and more informative updates in complex learning problems [21–24]. Related studies have also shown that Gauss–Newton-type and inexact second-order procedures can improve optimization performance in classification settings, including neural and binary classification problems [23–25]. However, explicit investigation of Hessian-free optimization in educational data mining remains limited.

Against this background, this study addresses three research questions. First, does the combination of academic and non-academic features yield better graduation predicate prediction performance than academic features alone? Second, can Hessian-free optimization outperform standard first-order optimizers across academic-only, non-academic-only, and combined feature groups? Third, when only non-academic features are available, can they be used to identify students who are potentially at risk of weak final academic achievement?

To address these questions, this study formulates two related prediction tasks. The first is a four-class classification problem using the final graduation predicate categories *Sufficient*, *Satisfactory*, *Very Satisfactory*, and *Cum Laude*. The second is a binary early risk-screening problem in which *Sufficient* is treated as the positive risk class and all remaining predicates are treated as the non-risk class. A deep neural network implemented as a multilayer perceptron is trained under four optimization methods, namely SGD with momentum, RMSProp, Adam, and Hessian-free optimization, so that the comparison focuses on feature-group and optimizer effects rather than architectural variation.

The contribution of this study is twofold. First, it integrates feature-group comparison and optimizer comparison within a unified deep neural classification framework for graduation predicate prediction. Second, it extends the analysis beyond multiclass graduation predicate classification by reformulating the problem as an early risk-screening task suitable for newly admitted students, for whom semester-based academic variables are not yet available. In this way, the study contributes methodologically by examining the role of Hessian-free optimization in deep neural classification and practically by identifying which non-academic variables can support early screening.

The remainder of this article is organized as follows. Section 2 presents the dataset representation, preprocessing steps, neural-network formulation, optimization procedures, and evaluation protocol. Section 3 reports the experimental results and discusses their implications. Section 4 concludes the paper.

2. Methods

This section presents the experimental and computational framework adopted in the study. It begins with the dataset notation, feature-group definitions, and target formulations. It then describes preprocessing and data partitioning before presenting the deep neural network architecture, the objective functions, and the training procedures. Particular attention is given to the damped Hessian-free method and its conjugate-gradient inner solver. The section concludes with the evaluation metrics and the additional subset analysis for non-academic early risk screening.

2.1. Notation and experimental setting

Let

$$\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$$

denote the dataset, where $\mathbf{x}_i$ is the predictor vector of the ith student and y_i is the corresponding target label. The predictor vector is partitioned into two groups,

$$\mathbf{x}_i = \left(\mathbf{x}_i^{(a)}, \mathbf{x}_i^{(n)}\right),$$

where $\mathbf{x}_i^{(a)}$ denotes academic features and $\mathbf{x}_i^{(n)}$ denotes non-academic features.

The academic feature group is defined as

$$\mathbf{x}_i^{(a)} = (\mathrm{GPA}_1, \mathrm{GPA}_2, \mathrm{GPA}_3, \mathrm{GPA}_4),$$

where GPA_s denotes the semester GPA in semester s. The non-academic feature vector is defined by

$$\begin{aligned}\mathbf{x}_i^{(n)} = (&\text{Gender, School Type, Boarding Experience,}\\ &\text{Admission Path, Arabic Proficiency,}\\ &\text{English Proficiency, Computer Proficiency,}\\ &\text{Major})^{\top}.\end{aligned} \tag{1}$$

Accordingly, three feature-group scenarios are considered:

$$\mathcal{X}_{\mathrm{acad}} = \mathbf{x}^{(a)}, \tag{2}$$

$$\mathcal{X}_{\mathrm{non}} = \mathbf{x}^{(n)}, \tag{3}$$

$$\mathcal{X}_{\mathrm{comb}} = \left(\mathbf{x}^{(a)}, \mathbf{x}^{(n)}\right). \tag{4}$$

Equations (2)–(4) define the three experimental settings used throughout the study, while Eq. (1) specifies the non-academic feature vector.

Two target formulations are used. In the multiclass task, the target space is

$$\mathcal{Y}_{\mathrm{multi}} = \{1, 2, 3, 4\},$$

corresponding to the ordered predicates

$$\{\textit{Sufficient},\ \textit{Satisfactory},\ \textit{Very Satisfactory},\ \textit{Cum Laude}\}.$$

In the binary risk-screening task, the target is defined by

$$y_i^{(r)} = \begin{cases} 1, & \text{if student } i \text{ has predicate } \textit{Sufficient},\\ 0, & \text{otherwise.}\end{cases} \tag{5}$$

Thus, by Eq. (5), $y_i^{(r)} = 1$ represents a student who is operationally regarded as academically at risk.

2.2. Data preprocessing

The preprocessing stage was designed according to the measurement scale of each feature. Empty strings were first converted into missing values. Missing entries in categorical variables were imputed using the mode, whereas missing entries in numerical variables were imputed using the median.

Let x_{ij} denote the value of feature j for sample i. For numerical variables, z-score standardization was applied:

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}, \tag{6}$$

where μ_j and σ_j are the mean and standard deviation of feature j computed from the training set. Eq. (6) ensures that numerical variables are placed on a comparable scale before training.

Categorical variables were transformed using one-hot encoding. Hence, the final input vector to the classifier can be written as

$$\tilde{\mathbf{x}}_i = T(\mathbf{x}_i),$$

where $T(\cdot)$ denotes the preprocessing operator composed of categorical expansion and, when appropriate, numerical standardization. The transformation T was fitted only on the training set and then applied to the validation and test sets to avoid information leakage.

2.3. Train–validation–test split

The dataset was divided into three disjoint subsets,

$$\mathcal{D} = \mathcal{D}_{\text{train}} \cup \mathcal{D}_{\text{val}} \cup \mathcal{D}_{\text{test}},$$

with proportions 60%, 20%, and 20%, respectively. The partition was produced through a two-stage stratified split. First, the full dataset was split into temporary training data (80%) and test data (20%). Second, the temporary training data were split into training data (75%) and validation data (25%), which yields the final 60–20–20 ratio. Stratification was performed with respect to the target labels so that class proportions were approximately preserved across subsets.

2.4. Deep neural network formulation

All experiments used the same deep neural network so that the comparison isolates the effects of feature groups and optimization methods. Let $\tilde{\mathbf{x}} \in \mathbb{R}^p$ denote the preprocessed input vector. The network defines a parametric mapping

$$f_\theta : \mathbb{R}^p \to \mathbb{R}^K,$$

where θ denotes all trainable weights and biases, and $K = 4$ for the multiclass task and $K = 1$ for the binary task.

The hidden representation is computed by three affine transformations followed by nonlinear activation:

$$\mathbf{u}^{(1)} = W^{(1)}\tilde{\mathbf{x}} + \mathbf{b}^{(1)}, \tag{7}$$
$$\mathbf{h}^{(1)} = \phi(\mathbf{u}^{(1)}), \tag{8}$$
$$\mathbf{u}^{(2)} = W^{(2)}\mathbf{h}^{(1)} + \mathbf{b}^{(2)}, \tag{9}$$
$$\mathbf{h}^{(2)} = \phi(\mathbf{u}^{(2)}), \tag{10}$$
$$\mathbf{u}^{(3)} = W^{(3)}\mathbf{h}^{(2)} + \mathbf{b}^{(3)}, \tag{11}$$
$$\mathbf{h}^{(3)} = \phi(\mathbf{u}^{(3)}). \tag{12}$$

Here, $\phi(t) = \max(0, t)$ is the rectified linear unit (ReLU). Equations (7)–(12) describe the feedforward transformation through the three hidden layers. The hidden-layer sizes were fixed at

$$(19, 12, 9).$$

During training, dropout with rate $r = 0.3$ was applied after each hidden layer. If $\mathbf{m}^{(\ell)}$ denotes a Bernoulli mask for layer ℓ, the dropout-transformed hidden state can be written as

$$\tilde{\mathbf{h}}^{(\ell)} = \mathbf{m}^{(\ell)} \odot \mathbf{h}^{(\ell)}, \tag{13}$$

where $\odot$ denotes elementwise multiplication. Eq. (13) formalizes the regularization mechanism used to reduce overfitting.

For the multiclass task, the output layer computes logits

$$\mathbf{a} = W^{(4)}\mathbf{h}^{(3)} + \mathbf{b}^{(4)},$$

and predicted class probabilities are obtained by the softmax transformation

$$\hat{y}_{ik} = \frac{\exp(a_{ik})}{\sum_{j=1}^{4} \exp(a_{ij})}, \qquad k = 1, 2, 3, 4. \tag{14}$$

Thus, Eq. (14) maps the logits into class probabilities that sum to one.

For the binary risk-screening task, the output is

$$a_i = W^{(4)}\mathbf{h}^{(3)} + \mathbf{b}^{(4)},$$

followed by the sigmoid transformation

$$\hat{y}_i = \sigma(a_i) = \frac{1}{1 + \exp(-a_i)}. \tag{15}$$

Eq. (15) gives the estimated probability that student i belongs to the risk class.

2.5. Objective functions

For the multiclass task, the categorical cross-entropy is defined as

$$\mathcal{L}_{\text{multi}}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{4} y_{ik} \log \hat{y}_{ik}, \tag{16}$$

where y_{ik} is the one-hot representation of the true class of sample i. Eq. (16) is minimized when the softmax probabilities in Eq. (14) align with the true multiclass labels.

For the binary risk-screening task, the binary cross-entropy is defined as

$$\mathcal{L}_{\text{bin}}(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i^{(r)} \log \hat{y}_i + (1 - y_i^{(r)}) \log(1 - \hat{y}_i)\right]. \tag{17}$$

In Eq. (17), $\hat{y}_i$ is obtained from the sigmoid model in Eq. (15).

For first-order optimizers, class weighting was used to reduce the influence of class imbalance. Let ω_c denote the weight for class c. Then the weighted loss can be written in the generic form

$$\mathcal{L}_w(\theta) = \frac{1}{N}\sum_{i=1}^{N} \omega_{y_i}\, \ell_i(\theta), \tag{18}$$

where $\ell_i(\theta)$ denotes the sample-wise loss term. Eq. (18) was used for the first-order baselines, whereas the Hessian-free implementation was kept unweighted in accordance with the computational design adopted in the experiments.

2.6. First-order optimization methods

Three first-order optimizers were used as baselines: SGD with momentum, RMSProp, and Adam. For SGD with momentum, the update rule is

$$\mathbf{v}_{t+1} = \mu \mathbf{v}_t - \eta \nabla \mathcal{L}(\theta_t), \tag{19}$$

$$\theta_{t+1} = \theta_t + \mathbf{v}_{t+1}, \tag{20}$$

where η is the learning rate and μ is the momentum coefficient. Equations (19) and (20) show that SGD with momentum uses only first-order information, augmented by an exponential moving average of past gradients. RMSProp and Adam were also employed as adaptive gradient-based optimizers. These methods likewise rely on first-order information and therefore provide suitable baselines for comparison with Hessian-free optimization.

2.7. Hessian-free optimization

Hessian-free optimization is a damped second-order method that uses curvature information without explicitly constructing the Hessian matrix [18]. Let

$$\mathbf{g}_t = \nabla \mathcal{L}(\theta_t)$$

denote the gradient at iteration t. In a local quadratic approximation around θ_t, the objective is approximated by

$$m_t(\mathbf{p}) = \mathcal{L}(\theta_t) + \mathbf{g}_t^\top \mathbf{p} + \frac{1}{2}\mathbf{p}^\top B_t \mathbf{p}, \tag{21}$$

where $\mathbf{p}$ is a candidate step direction and B_t is a curvature matrix, implicitly represented through Hessian–vector products. Thus, Eq. (21) provides the local model used to determine a Newton-type search direction.

To stabilize the step, a damping term is introduced, and the search direction $\mathbf{p}_t$ is obtained by approximately solving

$$A_t \mathbf{p}_t = -\mathbf{g}_t, \qquad A_t = B_t + \lambda_t I, \tag{22}$$

where $\lambda_t > 0$ is the damping parameter. Eq. (22) is the core linear system of the Hessian-free iteration.

The matrix A_t is never formed explicitly. Instead, the method computes products of the form

$$\mathbf{v} \mapsto A_t \mathbf{v} = (B_t + \lambda_t I)\mathbf{v}, \tag{23}$$

using automatic differentiation. Eq. (23) is implemented through Hessian–vector products, following the technique introduced by Pearlmutter [19], which makes second-order information available at a computational cost comparable to gradient evaluation.

After an approximate solution $\mathbf{p}_t$ is obtained, the parameter update is

$$\theta_{t+1} = \theta_t + \alpha_t \mathbf{p}_t, \tag{24}$$

where α_t is determined by backtracking. The quality of the step is then measured by

$$\rho_t = \frac{\mathcal{L}(\theta_t) - \mathcal{L}(\theta_t + \alpha_t \mathbf{p}_t)}{-\mathbf{g}_t^\top \mathbf{p}_t - \frac{1}{2}\mathbf{p}_t^\top A_t \mathbf{p}_t}. \tag{25}$$

Equations (24) and (25) define the accepted parameter step and the agreement ratio between predicted and realized reduction. If ρ_t is large, the damping parameter is reduced; if ρ_t is small, the damping parameter is increased. This rule adapts the trust in the local quadratic model. Such a curvature-aware perspective remains relevant in current neural optimization research, where exact or approximate second-order schemes continue to be studied as viable alternatives to purely first-order training strategies [22–24].

2.8. Conjugate gradient as the inner solver

The linear system in Eq. (22) was solved approximately by the conjugate gradient (CG) method, which is well suited for large symmetric positive definite systems and requires only matrix–vector products [20]. Since damping makes $A_t = B_t + \lambda_t I$ more numerically stable, CG can be applied efficiently within each Hessian-free iteration.

Let

$$A = A_t, \qquad \mathbf{b} = -\mathbf{g}_t.$$

Starting from an initial approximation $\mathbf{x}_0 = \mathbf{0}$, the CG iterations are defined by

$$\mathbf{r}_0 = \mathbf{b} - A\mathbf{x}_0, \tag{26}$$

$$\mathbf{p}_0 = \mathbf{r}_0. \tag{27}$$

Then, for $k = 0, 1, 2, \ldots$, the updates are

$$\alpha_k = \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\mathbf{p}_k^\top A \mathbf{p}_k}, \tag{28}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k, \tag{29}$$

$$\mathbf{r}_{k+1} = \mathbf{r}_k - \alpha_k A \mathbf{p}_k, \tag{30}$$

$$\beta_{k+1} = \frac{\mathbf{r}_{k+1}^\top \mathbf{r}_{k+1}}{\mathbf{r}_k^\top \mathbf{r}_k}, \tag{31}$$

$$\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_{k+1} \mathbf{p}_k. \tag{32}$$

Equations (26)–(32) define the inner iterative solver used to approximate the solution of Eq. (22). The final approximation $\mathbf{x}_{k+1}$ is then used as the Hessian-free search direction $\mathbf{p}_t$.

Algorithm 1 summarizes the complete damped Hessian-free training procedure used in this study, including the CG inner solver, backtracking line search, and damping adaptation.

2.9. Evaluation metrics

Performance was assessed on the held-out test set using accuracy, precision, recall, and F1-score. For the binary task, these are defined by

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{33}$$

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{34}$$

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{35}$$

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{36}$$

Equations (33)–(36) were used to evaluate the binary risk-screening task. For the multiclass task, weighted precision, weighted recall, and weighted F1-score were used. Confusion matrices were also inspected to analyse class-level behaviour. In addition, training and validation loss curves and accuracy curves were recorded to compare convergence across optimizers.

Algorithm 1 Damped HFO with conjugate gradient

Require: Initial parameter θ_0, initial damping λ_0, maximum CG iterations M, backtracking factor $\beta \in (0,1)$

for $t = 0, 1, 2, \ldots$ until convergence **do**
  Compute gradient $\mathbf{g}_t = \nabla \mathcal{L}(\theta_t)$
  Define the linear operator $A_t \mathbf{v} = (B_t + \lambda_t I)\mathbf{v}$
  Approximately solve $A_t \mathbf{p}_t = -\mathbf{g}_t$ using M steps of conjugate gradient
  Compute the predicted reduction

$$\Delta_t^{\text{pred}} = -\mathbf{g}_t^\top \mathbf{p}_t - \frac{1}{2}\mathbf{p}_t^\top A_t \mathbf{p}_t$$

  Set $\alpha_t \leftarrow 1$
  while $\mathcal{L}(\theta_t + \alpha_t \mathbf{p}_t) \geq \mathcal{L}(\theta_t)$ and α_t is sufficiently large **do**
    $\alpha_t \leftarrow \beta \alpha_t$
  end while
  Update $\theta_{t+1} \leftarrow \theta_t + \alpha_t \mathbf{p}_t$
  Compute

$$\rho_t = \frac{\mathcal{L}(\theta_t) - \mathcal{L}(\theta_{t+1})}{\Delta_t^{\text{pred}}}$$

  if $\rho_t > 0.75$ **then**
    Decrease damping parameter λ_t
  else if $\rho_t < 0.25$ **then**
    Increase damping parameter λ_t
  end if
end for
return Final parameter θ

2.10. Non-academic subset analysis for early screening

To determine which non-academic variables are most useful for screening newly admitted students, an additional subset analysis was conducted. Let

$$\mathcal{S} \subseteq \mathcal{X}_{\text{non}}$$

be a candidate subset of non-academic features. For each subset $\mathcal{S}$, a binary classifier

$$f_\theta^{(\mathcal{S})} : \mathcal{S} \to \{0, 1\}$$

was trained using the risk definition in Eq. (5). The evaluated subsets included single-feature models, small combinations, readiness-related subsets, admission-background subsets, and the complete non-academic feature set. This procedure was intended to identify which non-academic variables, individually or jointly, provide the strongest predictive signal for early academic risk screening.

2.11. Implementation environment

The experiments were implemented in Python using TensorFlow/Keras for deep neural network modelling and scikit-learn for preprocessing, data splitting, class-weight estimation, and evaluation. The same random seed was used throughout the experiments to improve reproducibility. The use of deep neural classification in this study is consistent with recent educational data mining and deep-learning research that applies supervised learning models to student performance and risk-prediction tasks [2, 3].

3. Results and Discussion

This section presents the experimental findings for both the multiclass graduation predicate classification task and the binary early risk-screening task. It begins with a comparison of feature-group performance, then evaluates the role of the optimizer, and finally discusses class-level behaviour and admission-time risk screening based on non-academic variables.

3.1. Multiclass classification performance across feature groups

The multiclass experiments compared three feature-group scenarios, namely academic-only, non-academic-only, and combined academic–non-academic features. For each scenario, the same multilayer perceptron architecture was trained using SGD with momentum, RMSProp, Adam, and Hessian-free optimization.

Table 1: Best classification performance for each feature-group scenario

Feature group	Best optimizer	Accuracy	Precision	Recall	F1-score
Academic-only	HFO	0.8434	0.8116	0.8434	0.8232
Non-academic-only	HFO	0.6111	0.5922	0.6111	0.5623
Combined	HFO	0.8478	0.8170	0.8478	0.8274

As shown in Table 1, the combined feature group achieved the best overall multiclass classification performance, with an accuracy of 0.8478 and a weighted F1-score of 0.8274. The academic-only scenario followed closely, with an accuracy of 0.8434 and a weighted F1-score of 0.8232. By contrast, the non-academic-only scenario remained substantially weaker, reaching an accuracy of 0.6111 and a weighted F1-score of 0.5623.

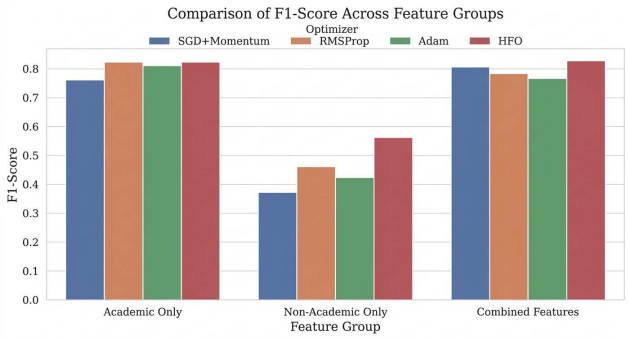

Fig. 1: Comparison of weighted F1-scores across academic-only, non-academic-only, and combined feature-group scenarios under the four optimization methods. Hessian-free optimization achieved the strongest F1-score in all scenarios, while the combined feature group produced the best overall result.

Fig. 1 summarizes the multiclass results visually. Two patterns are immediately clear. First, Hessian-free optimization produced the highest weighted F1-score in all three feature-group scenarios. Second, the combined academic–non-academic representation slightly outperformed the academic-only setting, whereas the non-academic-only scenario remained much weaker.

These results indicate that combining academic and non-academic variables yields the best predictive representation.

However, the improvement of the combined model over the academic-only model is relatively small. The appropriate interpretation is therefore not that non-academic variables radically transform graduation predicate classification, but rather that they provide complementary information that slightly improves the predictive value of academic indicators. Academic variables remain the dominant source of information for final predicate classification, whereas non-academic variables contribute an additional but modest gain. This pattern is consistent with literature showing that academic indicators often remain the strongest predictors of student outcomes, while demographic, admission, and readiness-related variables can contribute supplementary information, especially in early-stage prediction settings [2, 11].

At the same time, the non-academic-only scenario still achieved performance that was better than naive prediction. This indicates that non-academic variables contain useful predictive signal even in the absence of semester achievement indicators. Such evidence is important because it suggests that admission-time information is not irrelevant for academic prediction, even though it is not as strong as academic performance data.

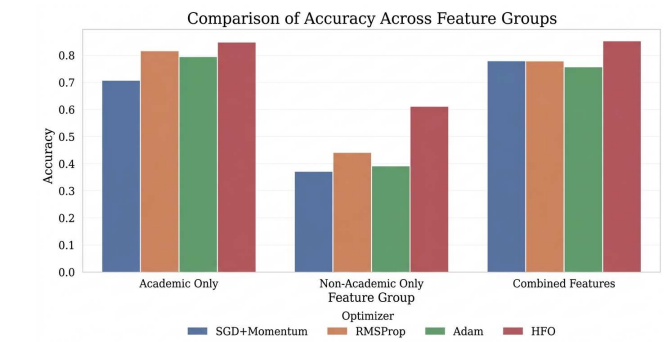

Fig. 2: Comparison of classification accuracy across academic-only, non-academic-only, and combined feature-group scenarios under the four optimization methods. The same ranking observed for weighted F1-score is also visible in accuracy, with HFO achieving the strongest performance across all scenarios.

The same ranking is also visible in Fig. 2, which compares accuracy across scenarios and optimizers. The consistency between Fig. 1 and Fig. 2 strengthens the conclusion that the superiority of HFO and the combined feature representation is not limited to a single evaluation metric.

3.2. Comparison of optimization methods

To examine the role of the optimizer more directly, each feature-group scenario was evaluated under four training methods: SGD with momentum, RMSProp, Adam, and Hessian-free optimization.

Table 2: Optimizer comparison for the academic-only feature group

Optimizer	Accuracy	Precision	Recall	F1-score
SGD+Momentum	0.5714	0.8065	0.5714	0.6516
RMSProp	0.8124	0.8466	0.8124	0.8221
Adam	0.8081	0.8436	0.8081	0.8193
HFO	0.8434	0.8116	0.8434	0.8232

Table 3: Optimizer comparison for the non-academic-only feature group

Optimizer	Accuracy	Precision	Recall	F1-score
SGD+Momentum	0.3831	0.8122	0.3831	0.2378
RMSProp	0.4430	0.5188	0.4430	0.4609
Adam	0.5585	0.6032	0.5585	0.3185
HFO	0.6111	0.5922	0.6111	0.5623

Table 4: Optimizer comparison for the combined feature group

Optimizer	Accuracy	Precision	Recall	F1-score
SGD+Momentum	0.7785	0.8064	0.7785	0.8056
RMSProp	0.8124	0.8238	0.8124	0.8166
Adam	0.8110	0.7954	0.8110	0.7943
HFO	0.8478	0.8170	0.8478	0.8274

Tables 2–4 show that Hessian-free optimization was the strongest optimizer in all three feature-group scenarios. In the academic-only setting, HFO achieved the highest weighted F1-score of 0.8232, only slightly above the best first-order baseline, namely RMSProp with 0.8221. In the non-academic-only setting, HFO produced a much stronger result than the first-order baselines, with an F1-score of 0.5623 compared with 0.4609 for RMSProp, 0.3185 for Adam, and 0.2378 for SGD with momentum. In the combined-feature scenario, HFO again achieved the best overall performance, with an F1-score of 0.8274, exceeding SGD with momentum (0.8056), RMSProp (0.8166), and Adam (0.7943).

Table 5 makes the advantage of HFO more explicit by comparing it directly with the strongest first-order optimizer in each scenario. The gain in the academic-only setting is small but positive, indicating that HFO remained competitive even when academic predictors already dominated the classification task. The largest gain appears in the non-academic-only scenario, where HFO improved the weighted F1-score by 0.1013, corresponding to a relative gain of 21.98%. In the combined-feature scenario, HFO also produced a clear improvement over the best first-order baseline, with an absolute F1-score gain of 0.0218.

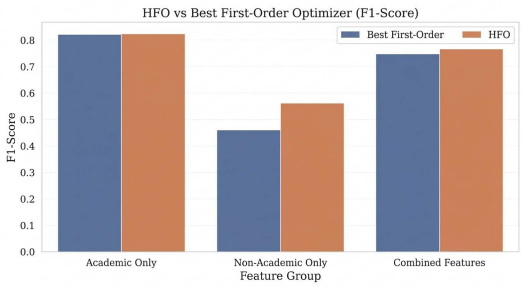

Fig. 3: Comparison between HFO and the strongest first-order optimizer in each multiclass scenario based on weighted F1-score. The most substantial advantage of HFO appears in the non-academic-only feature group.

Fig. 3 highlights the same pattern visually. HFO consistently outperformed the strongest first-order baseline across all feature-group scenarios. However, the magnitude of the gain was not uniform. In the academic-only setting the advantage was marginal, in the combined setting it was moderate, and in the non-academic-only setting it was substantial. This suggests that curvature-informed optimization is especially

Table 5: Performance gain of HFO over the best first-order optimizer in each multiclass scenario

Scenario	Best first-order	Best first-order F1	HFO F1	Absolute gain	Relative gain (%)
Academic-only	RMSProp	0.8221	0.8232	0.0010	0.13
Non-academic-only	RMSProp	0.4609	0.5623	0.1013	21.98
Combined	SGD+Momentum	0.8056	0.8274	0.0218	2.71

beneficial when the classifier must learn from a feature space dominated by encoded categorical admission variables rather than from stronger semester-based academic predictors.

Table 6: Mean optimizer performance across the three multiclass scenarios

Optimizer	Mean accuracy	Mean precision	Mean recall	Mean F1-score
SGDM	0.6183	0.7341	0.6183	0.6464
RMSProp	0.6777	0.7464	0.6777	0.6890
Adam	0.6484	0.7372	0.6484	0.6666
HFO	0.7674	0.7321	0.7674	0.7376

Table 6 further strengthens the argument by aggregating performance across scenarios. HFO achieved the highest mean accuracy (0.7674), mean recall (0.7674), and mean weighted F1-score (0.7376), clearly above the three first-order methods. This shows that the superiority of HFO is not limited to a single scenario, but persists across all multiclass feature-group settings.

Fig. 4 complements the metric-based comparison by revealing the training dynamics in the most informative scenario, namely the combined feature setting. The HFO-based model showed the strongest final validation accuracy and the lowest final validation loss, indicating not only strong final performance but also favourable optimization behaviour. In contrast, the first-order methods converged to weaker validation performance, with SGD with momentum showing the least competitive trajectory. This dynamic behaviour is consistent with the metric-based comparison and supports the argument that HFO is a more effective training strategy for the present classification problem.

Taken together, the results in Tables 2–6 and Figures 3–4 support the usefulness of Hessian-free optimization for deep neural classification in educational datasets. Because HFO incorporates curvature information through Hessian–vector products and solves the corresponding damped Newton-type system using conjugate gradient, it can produce more informative update directions than gradient-only methods. The consistency of its superiority across all feature groups, together with its clear gains over the strongest first-order baselines, constitutes the main computational contribution of this study.

3.3. Class-level behaviour of the best multiclass model

To examine the behaviour of the best-performing multiclass classifier at the class level, the confusion matrix of the combined-feature HFO model was analysed.

Fig. 5 shows that the combined HFO model performed strongly on the dominant predicate categories, particularly *Very Satisfactory* and *Cum Laude*. However, the weakest class, namely *Sufficient*, remained difficult to identify correctly. In the confusion matrix, the few samples in this class were still mapped into a higher predicate category.

Fig. 5: Confusion matrix of the HFO-based classifier for the combined academic–non-academic feature scenario. The model performed strongly on the dominant predicate classes, but the lowest predicate class (*Sufficient*) remained difficult to identify.

This observation qualifies the interpretation of the strong overall multiclass metrics. The model achieved high accuracy and weighted F1-score largely because it classified the dominant classes effectively. It also helps explain why the subsequent binary risk-screening task is highly challenging. Since *Sufficient* is extremely rare in the dataset, the model has limited opportunity to learn a robust boundary for this minority class. Therefore, the multiclass task is substantially easier for the dominant predicate groups than for the weakest academic group.

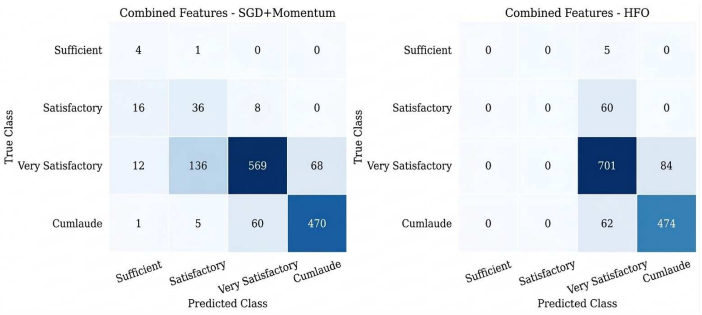

Fig. 6: Confusion-matrix comparison for the combined feature scenario between the strongest first-order optimizer and HFO. HFO improves the overall distribution of correct predictions in the dominant classes, although minority-class recovery remains limited.

Fig. 6 compares HFO directly with the strongest first-order baseline in the combined-feature scenario. The comparison shows that HFO yields a better overall allocation of predictions in the dominant classes and thereby supports the metric-based evidence that it is the strongest optimizer in this setting. At the same time, both matrices confirm that minority-class recovery remains a fundamental challenge, which is mainly attributable to the extreme scarcity of the *Sufficient* class rather than to optimization alone.

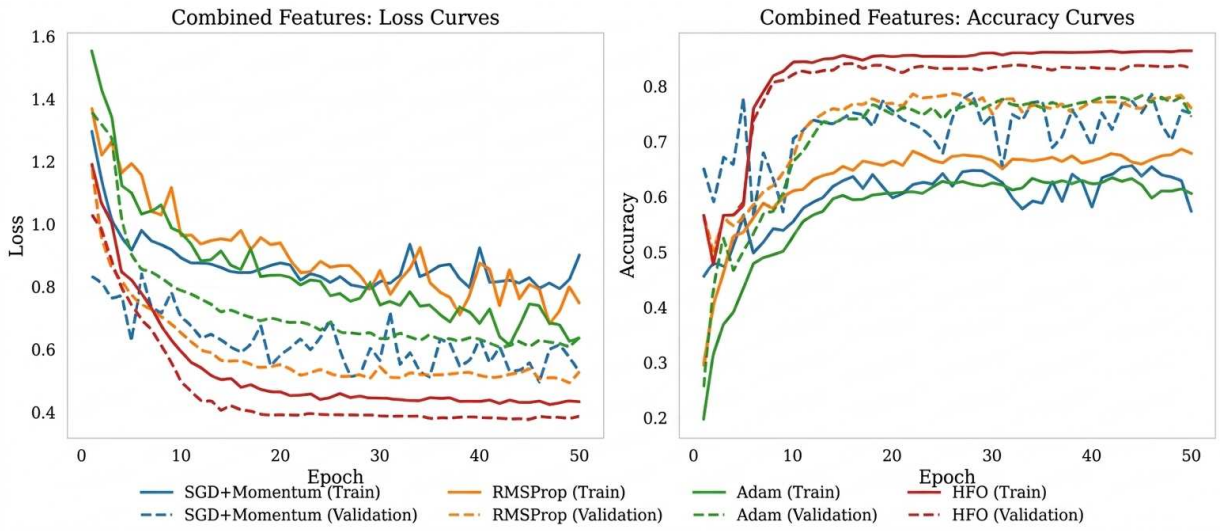

Fig. 4: Training and validation loss and accuracy curves for the combined academic–non-academic feature scenario. Hessian-free optimization reached the strongest final validation accuracy and the lowest final validation loss among the evaluated optimizers.

3.4. Binary early risk screening based on non-academic variables

As an additional analysis, the problem was reformulated as a binary early risk-screening task. In this setting, students with the final predicate *Sufficient* were labelled as positive cases and all remaining students were treated as negative cases. Because the purpose of this analysis was to simulate an admission-time screening setting, only non-academic variables were used.

The risk-screening results must be interpreted carefully. The dummy baseline achieved an accuracy of 0.9964 but a recall of 0.0000 and an F1-score of 0.0000, which indicates that the positive class was extremely rare. Therefore, accuracy is not an informative metric for this screening task. A classifier that simply predicts the majority class can appear highly accurate while failing completely to identify at-risk students.

For this reason, recall and F1-score are more informative than raw accuracy in Table 7. From this perspective, the non-academic models did show useful signal. The best overall subset, namely the full non-academic feature set including gender, achieved a recall of 0.4000 and an F1-score of 0.0252, which is clearly superior to the dummy baseline. Likewise, the best single feature, *Major*, achieved a recall of 0.6000 and an F1-score of 0.0242, indicating that it alone contains informative structure for identifying students who may later fall into the *Sufficient* category.

Nevertheless, the absolute performance of the screening models remained limited. Although the non-academic subsets improved substantially over the dummy baseline in terms of recall and F1-score, the resulting F1-scores were still very small. This means that non-academic variables do contain predictive signal for early screening, but the screening problem remains difficult because of severe class imbalance and the limited information available before academic outcomes emerge. This interpretation is also consistent with the broader early-warning literature, in which models based on limited early-stage information can still provide useful screening value even when absolute predictive performance remains constrained [26, 27].

Accordingly, the correct interpretation is not that non-academic features can already provide highly accurate risk prediction. Rather, the findings show that they provide a limited but meaningful basis for early warning. In practical terms, they may support first-stage screening, but should not be interpreted as a fully reliable standalone decision instrument.

3.5. Single non-academic features for early screening

To identify the most informative individual non-academic predictors, single-feature models were compared.

Table 8: Best-performing single non-academic features for early risk screening

Feature	Best optimizer	Accuracy	Precision	Recall	F1-score
Major	RMSProp	0.8254	0.0123	0.6000	0.0242
Gender	RMSProp	0.7082	0.0083	0.8000	0.0164
Arabic Proficiency	RMSProp	0.0235	0.0041	0.8000	0.0082
Admission Path	SGDM	0.4941	0.0039	1.0000	0.0079
School Type	Adam	0.3993	0.0038	0.6000	0.0076
Boarding Experience	HFO	0.3262	0.0046	0.6000	0.0091
English Proficiency	HFO	0.4606	0.0042	0.6000	0.0084
Computer Proficiency	HFO	0.4385	0.0040	0.6000	0.0080

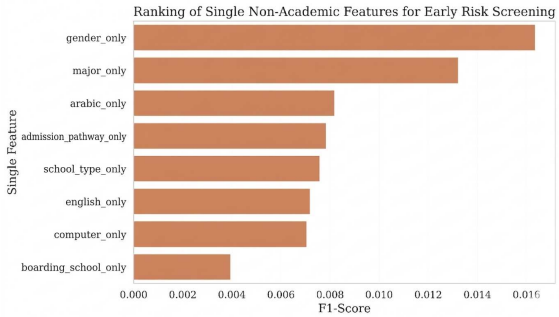

Fig. 7: Ranking of single non-academic features for early risk screening based on F1-score. *Major* emerged as the strongest individual predictor among the evaluated non-academic variables.

Among individual non-academic predictors, *Major* emerged as the strongest single feature, with an F1-score of 0.0242 and recall of 0.6000. Fig. 7 confirms this ranking visually

Table 7: Summary of binary early risk-screening performance

Category	Feature set	Best optimizer	Accuracy	Recall	F1-score
Best overall	all_nonacademic_with_gender	SGD+Momentum	0.8882	0.4000	0.0252
Best single feature	major_only	RMSProp	0.8254	0.6000	0.0242
Best multi-feature	all_nonacademic_with_gender	SGD+Momentum	0.8882	0.4000	0.0252
Dummy baseline	risk_dummy_baseline	DummyMostFrequent	0.9964	0.0000	0.0000

and shows that *Major* stands clearly above the other single-variable models. Substantively, this may reflect differences in curriculum demands, preparation profiles, or difficulty patterns across study programmes.

Although *Gender* produced the second-highest F1-score among single features, its overall predictive contribution remained modest, and its use in practical screening may require additional ethical consideration. Other variables such as *Admission Path*, *Arabic Proficiency*, *English Proficiency*, and *Computer Proficiency* also exhibited some predictive signal, but none matched the individual contribution of *Major*.

It is also noteworthy that some single-feature models achieved high recall but extremely low precision. For example, *Admission Path* reached recall 1.0000 with very low precision, indicating that the model identified nearly all positive cases at the cost of producing many false positives. Therefore, these variables may still be useful in a high-sensitivity screening context, but not as precise standalone predictors.

3.6. Multi-feature non-academic subsets for early screening

The next analysis examined combinations of non-academic variables in order to determine whether a compact subset could provide stronger practical screening performance.

Table 9: Best-performing multi-feature non-academic subsets for early risk screening

Feature subset	Best optimizer	Accuracy	Precision	Recall	F1-score
all_nonacademic_with_gender	SGDM	0.8882	0.0130	0.4000	0.0252
all_nonacademic_without_gender	Adam	0.8081	0.0112	0.6000	0.0221
core_screening	SGDM	0.8225	0.0106	0.5000	0.0210
major_admission	SGDM	0.8269	0.0105	0.5000	0.0208
extended_screening	SGDM	0.8232	0.0099	0.5000	0.0196

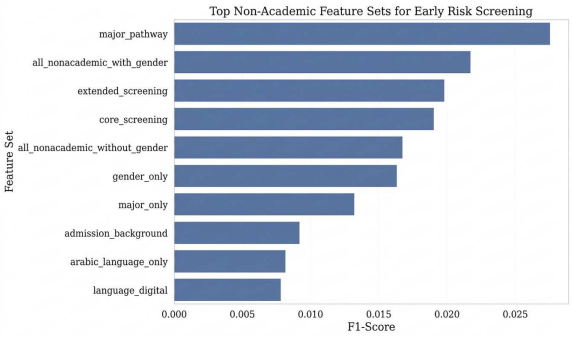

Fig. 8: Top non-academic feature subsets for early risk screening ranked by F1-score. The full non-academic set achieved the best overall screening performance, while smaller subsets centered on study-programme and admission-related variables remained competitive.

The full non-academic feature set including gender produced the best overall screening result, with an F1-score of 0.0252. However, the version without gender was only slightly worse, with an F1-score of 0.0221 and a higher recall of 0.6000. The difference between these two settings is therefore small. This suggests that, from a practical perspective, the exclusion of gender may be acceptable if institutional policy favours a more cautious and ethically conservative screening design.

Fig. 8 further shows that the highest-ranking subsets are not arbitrary combinations, but are dominated by models that include broad non-academic information or strong study-programme-related structure. Among the reduced subsets, the *core_screening* combination and the *major_admission* combination performed competitively, although neither surpassed the full feature set. This is relevant for implementation, because a faculty may prefer a smaller and more interpretable screening instrument when operational simplicity is important.

Overall, the subset analysis shows that the best screening performance is achieved not by a single variable alone, but by combining several non-academic variables. However, the gain from using all non-academic features remains limited in absolute terms, which again reflects the difficulty of the task under extreme class imbalance.

3.7. Implications for faculty-level early intervention

The findings of this study have two practical implications. First, once semester achievement variables are available, the best predictive strategy is to combine academic and non-academic information, ideally using Hessian-free optimization as the training method. This provides the most accurate basis for multiclass graduation predicate classification.

Second, when only admission-time variables are available, non-academic features can still support early risk screening, even though the resulting performance is limited. In this context, the main value of the screening model is not to deliver highly precise final predictions, but to function as an early-warning filter that identifies a subset of students who may benefit from additional mentoring, preparatory support, or closer academic monitoring.

The subset analysis further suggests that *Major* is the strongest single non-academic predictor, while broader combinations of non-academic variables yield the best overall performance. Therefore, if a faculty wishes to construct a simple but informative early screening tool, study-programme information should likely be treated as a core component.

3.8. Limitations

Several limitations should be acknowledged. First, the experiments were conducted on a single institutional dataset, so the results may not generalize directly to other universities or study programmes. Second, the evaluation used a fixed train–validation–test split rather than repeated resampling

or cross-validation. Third, the risk label was derived operationally from the lowest final predicate category rather than from formal dropout status. Therefore, the binary task should be interpreted as academic risk screening rather than direct dropout prediction.

A further limitation concerns the extreme rarity of the positive class in the binary screening experiments. Because of this imbalance, precision and F1-score remained low even for the best-performing subsets. Future work may therefore benefit from additional evaluation measures such as balanced accuracy, ROC-AUC, or PR-AUC, as well as data-level or algorithm-level imbalance-handling techniques.

3.9. Discussion summary

Overall, the experimental results support four main conclusions. First, the combined academic–non-academic representation produced the best multiclass classification performance, although its improvement over the academic-only representation was marginal. Second, Hessian-free optimization was consistently the strongest optimizer across academic-only, non-academic-only, and combined scenarios. Third, the clearest advantage of HFO over first-order optimization appeared in the non-academic-only scenario, where the gain over the strongest first-order baseline was substantial. Fourth, non-academic variables available at admission time do contain meaningful predictive signal for early screening of students who may later fall into the *Sufficient* category, but this screening performance remains limited because of severe class imbalance and the inherent difficulty of predicting long-term academic outcomes from pre-admission information alone.

4. Conclusion

This study examined student graduation predicate prediction and early academic risk screening using academic-only, non-academic-only, and combined academic–non-academic feature groups under four deep neural network optimization methods. The results show that the combined feature group achieved the best overall multiclass classification performance, although its advantage over the academic-only group was only marginal. This indicates that academic variables remain the strongest predictors, while non-academic variables contribute complementary information.

Hessian-free optimization consistently achieved the best performance across all feature-group scenarios. Its advantage over the strongest first-order optimizer was marginal in the academic-only setting, moderate in the combined setting, and substantial in the non-academic-only setting. In addition, HFO achieved the highest mean accuracy and mean weighted F1-score across the three multiclass scenarios. These findings support its effectiveness as a training strategy for deep neural classification in educational data.

For the binary early risk-screening task, non-academic variables alone provided meaningful but limited predictive signal for identifying students who may later fall into the *Sufficient* category. Among individual predictors, *Major* emerged as the strongest single non-academic feature, while the full non-academic feature set produced the best overall screening performance. These findings suggest that non-academic variables can support first-stage screening of newly admitted students, although they are not sufficient for highly accurate standalone risk prediction.

Future work may evaluate the proposed framework on broader institutional datasets, use repeated validation strategies, and incorporate imbalance-aware methods to improve early risk-screening performance.

CRediT Authorship Contribution Statement

Andy Irawan: Conceptualization, Methodology, Software, Data Curation, Formal Analysis, Visualization, Writing–Original Draft. **Zainal Abidin:** Validation, Investigation, Writing–Review & Editing, Supervision. **Mohammad Jamhuri:** Conceptualization, Methodology, Formal Analysis, Supervision, Writing–Review & Editing.

Declaration of Generative AI and AI-assisted technologies

Generative AI was used in a limited manner to assist with language refinement, structural editing, and manuscript drafting support.

Declaration of Competing Interest

The authors declare no competing interests.

Funding and Acknowledgments

This research received no external funding.

Data Availability

The data supporting the findings of this study are available from the corresponding author upon reasonable request and subject to institutional data-sharing restrictions and confidentiality considerations.

References

[1] Chaka Chaka. “Educational Data Mining, Student Academic Performance Prediction, Prediction Methods, Algorithms and Tools: An Overview of Reviews”. In: *Journal of e-Learning and Knowledge Society* 18.2 (2022), pp. 58–69. DOI: 10.20368/1971-8829/1135578.

[2] Bayan Alnasyan, Mohammed Basheri, and Madini O. Alassafi. “The Power of Deep Learning Techniques for Predicting Student Performance in Virtual Learning Environments: A Systematic Literature Review”. In: *Computers and Education: Artificial Intelligence* 6 (2024), p. 100231. DOI: 10.1016/j.caeai.2024.100231.

[3] Alexander E. J. Villegas-Espinoza and Jorge Isaac Necochea-Chamorro. “Using Deep Learning in Student Performance Prediction: A Systematic Review”. In: *TEM Journal* 14.3 (2025), pp. 2472–2482. DOI: 10.18421/TEM143-51.

[4] Wan Fatimah Wan Yaacob et al. “Supervised Data Mining Approach for Predicting Student Performance”. In: *Indonesian Journal of Electrical Engineering and Computer Science* 16.3 (2019), pp. 1584–1592. DOI: 10.11591/ijeecs.v16.i3.pp1584-1592.

[5] A. M. Shahiri, W. Husain, and N. A. Rashid. "A Review on Predicting Student's Performance Using Data Mining Techniques". In: *Procedia Computer Science* 72 (2015), pp. 414–422. DOI: 10.1016/j.procs.2015.12.157.

[6] Sri Widaningsih. "Perbandingan Metode Data Mining Untuk Prediksi Nilai dan Waktu Kelulusan Mahasiswa Prodi Teknik Informatika dengan Algoritma C4.5, Naïve Bayes, KNN dan SVM". In: *Jurnal Tekno Insentif* 13.1 (2019), pp. 16–25. DOI: 10.36787/jti.v13i1.78.

[7] A. Wibowo and A. Rohman. "Prediksi Predikat Kelulusan Mahasiswa Menggunakan Naive Bayes dan Decision Tree pada Universitas XYZ". In: *EXPERT: Jurnal Manajemen Sistem Informasi dan Teknologi* 12.2 (2022), p. 104. DOI: 10.36448/expert.v12i2.2810.

[8] M. A. Nurrohmat. "Aplikasi Pemrediksi Masa Studi dan Predikat Kelulusan Mahasiswa Informatika Universitas Muhammadiyah Surakarta Menggunakan Metode Naive Bayes". In: *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika* 1.1 (2015), pp. 29–34. DOI: 10.23917/khif.v1i1.1179.

[9] I. N. Rudy Hendrawan et al. "Klasifikasi Lama Studi dan Predikat Kelulusan Mahasiswa menggunakan Metode Naïve Bayes". In: *Jurnal Eksplora Informatika* 11.1 (2022), pp. 50–56. DOI: 10.30864/eksplora.v11i1.606.

[10] U. Indahyanti, N. L. Azizah, and H. Setiawan. "Pendekatan Ensemble Learning Untuk Meningkatkan Akurasi Prediksi Kinerja Akademik Mahasiswa". In: *Jurnal Sains Dan Informatika* 8.2 (2022), pp. 160–169. DOI: 10.34128/jsi.v8i2.459.

[11] Chayaporn Kaensar and Worayoot Wongnin. "Predicting New Student Performances and Identifying Important Attributes of Admission Data Using Machine Learning Techniques with Hyperparameter Tuning". In: *Eurasia Journal of Mathematics, Science and Technology Education* 19.12 (2023), em2369. DOI: 10.29333/ejmste/13863.

[12] Kam Cheong Li, Billy Tak-Ming Wong, and Thomas Hon-Tung Chan. "Predictive Analytics for University Student Admission: A Literature Review". In: *Blended Learning: Education in a Smart Learning Environment*. Springer, 2023, pp. 250–259. DOI: 10.1007/978-3-031-35731-2_22.

[13] Hanan Abdullah Mengash. "Using Data Mining Techniques to Predict Student Performance to Support Decision Making in University Admission Systems". In: *IEEE Access* 8 (2020), pp. 55462–55470. DOI: 10.1109/ACCESS.2020.2981905.

[14] Mustakim and Giantika Oktaviani. "Algoritma K-Nearest Neighbor Classification Sebagai Sistem Prediksi Predikat Prestasi Mahasiswa". In: *Jurnal Sains, Teknologi, dan Industri* 13.2 (2016), pp. 195–202. DOI: 10.24014/sitekin.v13i2.1688.

[15] N. B. Nasution et al. "Prediksi Lama Studi dan Predikat Kelulusan Mahasiswa Menggunakan Algoritma Supervised Learning". In: *G-Tech: Jurnal Teknologi Terapan* 7.2 (2023), pp. 386–395. DOI: 10.33379/gtech.v7i2.2077.

[16] La Ode Mohamad Zulfiqar, Nurul Renaningtias, and M. Y. Fathoni. "Educational Data Mining in Graduation Rate and Grade Predictions Utilizing Hybrid Decision Tree and Naïve Bayes Classifier". In: *Proceedings of the International Conferences on Information System and Technology 2019*. SCITEPRESS, 2020, pp. 151–157. DOI: 10.5220/0009907101510157.

[17] Francis Ofori, Elizaphan Maina, and Rhoda Gitonga. "Using Machine Learning Algorithms to Predict Students' Performance and Improve Learning Outcome: A Literature Based Review". In: *Journal of Information and Technology* 4.1 (2020), pp. 33–55. Available online.

[18] James Martens. "Deep Learning via Hessian-Free Optimization". In: *Proceedings of the 27th International Conference on Machine Learning*. 2010, pp. 735–742. DOI: 10.5555/3104322.3104416.

[19] Barak A. Pearlmutter. "Fast Exact Multiplication by the Hessian". In: *Neural Computation* 6.1 (1994), pp. 147–160. DOI: 10.1162/neco.1994.6.1.147.

[20] Jonathan Richard Shewchuk. *An Introduction to the Conjugate Gradient Method Without the Agonizing Pain*. Tech. rep. Carnegie Mellon University, 1994. Available online.

[21] Nitin Liladhar Rane et al. "Techniques and Optimization Algorithms in Deep Learning: A Review". In: *Applied Machine Learning and Deep Learning: Architectures and Techniques*. Deep Science Publishing, 2024, pp. 59–79. DOI: 10.70593/978-81-981271-4-3_3.

[22] Mikalai Korbit et al. "Exact Gauss-Newton Optimization for Training Deep Neural Networks". In: *Neurocomputing* 658 (2025), p. 131738. DOI: 10.1016/j.neucom.2025.131738.

[23] Mohammad Jamhuri et al. "Inexact Generalized Gauss–Newton–CG for Binary Cross-Entropy Minimization". In: *Jurnal Riset Mahasiswa Matematika* 5.2 (2025), pp. 102–122. DOI: 10.18860/jrmm.v5i2.34739.

[24] Mohammad Jamhuri et al. "Neural Networks Optimization via Gauss–Newton Based QR Factorization on SARS-CoV-2 Variant Classification". In: *Systems and Soft Computing* 7 (2025), p. 200195. DOI: 10.1016/j.sasc.2025.200195.

[25] Mohammad Jamhuri, Imam Mukhlash, and Mohammad Isa Irawan. "Performance Improvement of Logistic Regression for Binary Classification by Gauss-Newton Method". In: *Proceedings of the 2022 5th International Conference on Mathematics and Statistics*. 2022, pp. 12–16. DOI: 10.1145/3545839.3545842.

[26] David Bañeres et al. "An Early Warning System to Detect At-Risk Students in Online Higher Education". In: *Applied Sciences* 10.13 (2020), p. 4427. DOI: 10.3390/app10134427.

[27] Elizabeth Foster and Richard Siddle. "The Effectiveness of Learning Analytics for Identifying At-Risk Students in Higher Education". In: *Assessment & Evaluation in Higher Education* 45.6 (2020), pp. 842–854. DOI: 10.1080/02602938.2019.1682118.