

RESEARCH ARTICLE



A Multi-Model Approach to Academic Performance Prediction Using Student–Semester Educational Data

Tole Bi Yermek^{1,3} , Banu Yergesh^{1,2} , Amanzhol Yelemessov² , Madina Yelemessova² , Gulmira Bekmanova², Assel Omarbekova², Aizhan Nazyrova^{2,*} and Zhanar Lamasheva^{2,*}

¹Limited Liability Partnership “Tor Altai Nuclear Consulting”, Kazakhstan

²Institute of Digital Sciences and Artificial Intelligence, L.N. Gumilyov Eurasian National University, Kazakhstan

³Institute of Automation and Information Technology, Satbayev University, Kazakhstan

Abstract: Identifying students at academic risk early calls for forecasting that respects the temporal structure of educational data and stays readable for advising staff. In this work, linear, gradient-boosted, and deep tabular models are compared under one protocol on two tasks: next-semester performance prediction and binary risk classification. The data are aggregated student–semester records from a university Learning Management System, covering 732,965 academic records on 20,868 students over 14 academic years at a Central Asian institution. All models are evaluated under a strict temporal train–validation–test split, so each is asked to forecast a year it has never seen. The feature set adds two trajectory variables (avg_mark_lag1, delta_avg) for the direction of a student’s performance and two within-semester indicators (std_mark, min_mark) for its stability. On the 2022 test set, Ridge Regression reached $R^2 = 0.552$ and Logistic Regression PR-AUC = 0.432 (ROC-AUC = 0.904). The gradient-boosted ensembles (LightGBM, CatBoost, XGBoost) did not clearly beat the regularized linear baselines under temporal validation, and the deep tabular architectures (FT-Transformer, TabNet, AutoInt, MLP-embeddings, Deep & Cross), run under the identical protocol, did not outperform either family. A Top-K intervention rule recovers 33% of all at-risk students within the top 5% of ranked cases (lift $\approx 4.15\times$). That is a decision rule an advising office can work with under a fixed caseload.

Keywords: academic performance prediction, educational analytics, learning risk detection, machine learning, student-semester data

1. Introduction

UNESCO and OECD documents both address digital technologies and data analysis in education. UNESCO reports describe how educational data can track student progress and inform management decisions [1] and set out guidance for the use of AI in education and research [2]. OECD publications treat analytics as a way to support individual learning trajectories and reduce academic risk [3].

Most universities still work with retrospective measures such as end-of-semester GPA and formal reports. These describe final outcomes and offer little help in identifying learning difficulties at an early stage.

Machine learning and neural networks have been applied widely to academic performance prediction [4]. Several methodological problems persist, however. Random train–test splitting

ignores the temporal structure of educational data and inflates reported performance [5]. Target definitions and feature sets differ from study to study, which limits reproducibility and comparison [6]. Model classes are also rarely compared under one experimental protocol [4, 6].

Educational data raise difficulties of their own. The structure is heterogeneous, semesters depend on one another, samples are often small, and classes are imbalanced [7–9]. Predictions also have to be interpretable enough for teachers and administrators to act on [10, 11]. Together, these issues call for reproducible prediction frameworks that account for temporal structure and for how the output will be used.

This study predicts academic performance and identifies at-risk students from data aggregated at the student–semester level. The data are split chronologically to approximate real deployment conditions [5], and linear models, ensemble methods, and deep tabular architectures are evaluated under the same temporal protocol for both regression and classification [7].

Five contributions follow. The first is empirical: academic performance prediction is evaluated on 14 academic years of LMS data from a Central Asian higher-education institution, covering 20,868 students and 732,965 raw records. The second is

*Corresponding author: Aizhan Nazyrova, Institute of Digital Sciences and Artificial Intelligence, L.N. Gumilyov Eurasian National University, Kazakhstan. Email: nazyrova_aye_1@enu.kz. Zhanar Lamasheva, Institute of Digital Sciences and Artificial Intelligence, L.N. Gumilyov Eurasian National University, Kazakhstan. Email: lamasheva_zhb@enu.kz

the student–semester framework itself, which combines temporal validation with aggregated semester indicators and trajectory features (avg_mark_lag1, delta_avg). Third, regularized linear models, gradient-boosted ensembles, and deep tabular architectures are compared under a single temporal protocol. Fourth, a Top-K evaluation ranks students for intervention when advising resources are limited; delta_avg, std_mark, and min_mark are the signals an advisor works with. The fifth contribution is comparative: the framework is placed against existing prediction approaches, and Sections 3.1.1–3.1.2 and 4.1–4.7 describe its architecture together with the procedures for temporal validation, class imbalance, and interpretability.

The study addresses three research questions:

RQ1. To what extent do aggregated student–semester data predict academic outcomes for the following semester?

RQ2. How do regularized linear models, gradient-boosted ensembles, and deep tabular architectures compare under temporal validation?

RQ3. What role do delta_avg, std_mark, and min_mark play in identifying academic risk and selecting student support measures?

2. Literature Review

Research on educational data analysis concentrates on predicting academic performance and identifying students at risk. Digitalization has produced data volumes large enough for machine learning and analytics methods to be applied, in education as in other data-intensive domains [12, 13].

Recent reviews name academic performance prediction, student risk detection, and learning behavior analysis as the main areas of educational data mining [14]. Electronic records and assessment data can support predictive models even with fairly small feature sets [4], and both the educational context and the temporal structure of the data affect what learning analytics produces [6].

Work on at-risk students emphasizes early intervention [11]. How well such systems perform depends on the choice of

evaluation metrics and validation methods [6]. Limited data from the early stages of learning can still support a preliminary risk assessment, provided the model captures signs of declining performance [9].

Academic performance indicators and other educational features are used widely for prediction [15–18], and interpretability remains important for practical use [10, 19]. Deep learning for tabular educational data has drawn recent attention, including hybrid and attention-based models for performance and risk prediction [20]; the architectures themselves come from the general tabular-learning literature [21–23]. Tabular foundation models trained for in-context learning, such as TabICL [24], extend this direction to larger tabular datasets and offer an alternative to task-specific training. Systematic comparisons of neural architectures for structured data show that their effectiveness depends heavily on the validation protocol and the feature structure [21]. Attention-based architectures use attention mechanisms for feature selection [22], and recent studies compare attention with contrastive learning on tabular data [23]. With limited samples and class imbalance, however, deep models can be unstable [7].

Several studies underline the temporal structure of educational data. Forecasting the next semester requires shifting the target variable in time and abandoning random sampling [5]. More recent applied work reports far higher figures under random cross-validation (R^2 up to 0.989) [19], which shows how strongly the validation protocol shapes the performance a study can claim. Practice-oriented research addresses the integration of analytical models into educational information systems: AI-driven identification of at-risk students needs forecasting accuracy and explainability together [11]. Early warning systems are also treated as support for personalized intervention, where ranking students matters because institutional resources are limited [25]. Broad reviews confirm the steady interest in performance prediction while pointing to unresolved problems. There are no uniform temporal validation protocols, target variables are formalized differently across studies, and results are consequently hard to compare [17, 26]. Table 1 summarizes the key studies used in the review.

Table 1
Key works on predicting academic performance and the methods used

Ref.	Models used	Task	Data source
[21]	MLP, ResNet-like NN, FT-Transformer, TabTransformer, NODE, XGBoost, LightGBM	Tabular regression & classification (benchmark)	General tabular datasets (non-educational)
[22]	TabNet (attention-based neural network)	Tabular regression & classification	Structured tabular datasets
[27]	SAINT (row & column attention), SAINT + contrastive pretraining	Tabular regression & classification	Tabular datasets
[16]	Review of early, late and hybrid data-fusion strategies (classical ML and DL classifiers)	Academic performance prediction	Multimodal LMS, log, video, sensor and academic data (review)
[7]	MLP, TabNet, FT-Transformer, SAINT, AutoInt, NODE, Deep & Cross Network	Structured data regression & classification	Tabular datasets across domains (survey)
[11]	Decision Tree, Correlation-based Feature Selection (CFS), XAI	At-risk student detection	Academic, personality, and soft-skills data
[5]	Sequential campus-behavior classifier (SPC) with classical ML baselines	Next-term performance prediction	Short-term sequential campus-behavior records
[28]	Naive Bayes, Decision Tree, kNN, Random Forest, Logistic Regression, ANN (MLP)	Final-grade prediction	Moodle LMS logs, academic records

Recent research pays more attention to the temporal structure of educational data and to time-aware validation. Deep learning for tabular data is also being explored, although its performance can be sensitive to data quality and parameter settings. In early detection of academic risk, metrics that account for class imbalance and ranking are used more often, including PR-AUC and Recall at Top-K. Interpretability, meanwhile, remains an important requirement for practical application in educational management.

Educational analytics has produced a large body of work, but few papers use a single experimental approach that combines an aggregated semester data format, temporal separation of the sample, and a systematic comparison of linear, ensemble, and neural network models. The present study takes up that methodological gap through the student–semester format and a time-sensitive evaluation protocol.

Tables 2 and 3 situate the study within this literature. They summarize the results that recent academic performance and academic-risk studies report on their own datasets, including the authors’ prior deep learning model [20], and set the methodological choices of those studies beside the ones adopted here. The studies differ in datasets, targets, grading scales, and validation protocols, so their results are not directly comparable. The comparison therefore concerns methodological design and the main performance metric each study reports on its own data.

Two observations follow. The first concerns methodology. The present framework is conservative by design, and it is the only entry that combines aggregated student–semester data with a strict temporal train–validation–test split, so its metrics estimate

genuine next-semester generalization rather than within-cohort interpolation. Its headline figures ($R^2 = 0.552$; PR-AUC = 0.432) therefore sit below the ROC-AUC of 0.95 reported for the authors’ own behavioral OULAD model [20], which was evaluated under a standard split and had access to fine-grained weekly activity logs. The second observation concerns what the work contributes: a deployment-realistic protocol. Comparable predictive signal is recovered from academic records alone, under temporal validation, and it comes with an explicit Top-K intervention rule (33% of all at-risk students recovered within the top 5% of ranked cases; lift $\approx 4.15\times$) that most prior studies do not provide.

The framework differs from typical prior models along several further methodological dimensions, and these affect reliability and practical use most directly. Table 3 summarizes them.

3. Materials and methods

3.1. Proposed framework

This study proposes a student–semester academic-risk prediction framework: an end-to-end pipeline, built with deployment in mind, that turns raw Learning Management System (LMS) records into next-semester performance forecasts, calibrated risk probabilities and a ranked intervention list for academic advisors. The framework is the central contribution of this work. What it offers is a problem-specific methodological design rather than a new learning algorithm, defined by five coupled components, each of which addresses a recurring difficulty of educational data. In its operational form, the method is this pipeline instantiated

Table 2
Comparison of the proposed framework with recent academic performance prediction models

Ref.	Data and context	Validation	Task	Reported accuracy / headline metric (on its own data)
[5]	Short-term sequential campus-behavior data	Sequential / time-aware	Next-term performance prediction	Accuracy $\approx 86.9\%$, Recall $\approx 81.6\%$ (SPC sequence classifier)
[9]	Blended-learning course, early-stage limited data	Course-level holdout	Early at-risk detection	Prospective AUC/F1 close to the same-year benchmark for early at-risk detection
[11]	Academic and behavioral records (RADAR)	Standard holdout	At-risk early-warning detection	Accuracy $\approx 82.2\%$, F1 $\approx 83.3\%$ (RADAR; decision tree with feature selection)
[19]	Public multi-feature performance dataset	Random / cross-validation	Performance prediction with XAI	R^2 up to 0.989 on its primary dataset (0.772 on a harder set); ensemble voting regressor
[28]	Moodle LMS logs and academic records	Random / cross-validation	Final-grade prediction	F1 > 0.80 (k-NN and neural networks; 450 students, Moodle)
[20]	OULAD weekly behavioral activity (32 weeks) + static attributes	Standard split	At-risk classification (Bi-LSTM + MLP + temporal attention)	ROC-AUC 0.95; weighted F1 0.90; at-risk recall 0.79
This study	Aggregated student–semester LMS records (academic only); 20,868 students; 14 years	Strict temporal split (train $\leq$ 2020 / val 2021 / test 2022)	Next-semester regression + risk classification + Top-K intervention	R^2 0.552 (Ridge); ROC-AUC 0.904, PR-AUC 0.432, Recall@Top-5% 0.331 (Logistic); lift $\approx 4.15\times$

Table 3
Differences between the proposed framework and existing academic performance prediction models

Dimension	Typical existing prediction models	Proposed model
Data granularity	Course-, session- or click-level logs, or single-cohort snapshots	Aggregated student–semester data mart spanning 14 academic years and 20,868 students
Validation	Predominantly random or k-fold splits, which risk temporal leakage	Strict temporal train–validation–test split ($\leq 2020 / 2021 / 2022$) emulating deployment
Feature design	Large, often dataset-specific or behavioral feature sets	Compact, transferable trajectory and stability features (avg_mark_lag1, delta_avg, std_mark, min_mark)
Model scope	A single model class or a narrow comparison	Unified one-pipeline comparison of linear, gradient-boosted and deep tabular families
Target definition	Heterogeneous, study-specific targets	Two aligned, time-shifted targets (next-semester score and binary risk) from one definition
Output for practice	A probability or a class label	A Top-K ranked intervention list sized to advising capacity, with advisor-readable risk signals
Interpretability	Often post-hoc only (e.g., SHAP/LIME on opaque models)	Intrinsically interpretable trajectory indicators mapped to concrete support actions

with a regularized linear predictor: Ridge Regression for next-semester regression and Logistic Regression for risk classification. That instantiation is the configuration labeled “Proposed” in the controlled comparison of Section 4.7, where it is evaluated against standard baselines and recent state-of-the-art tabular methods under an identical protocol.

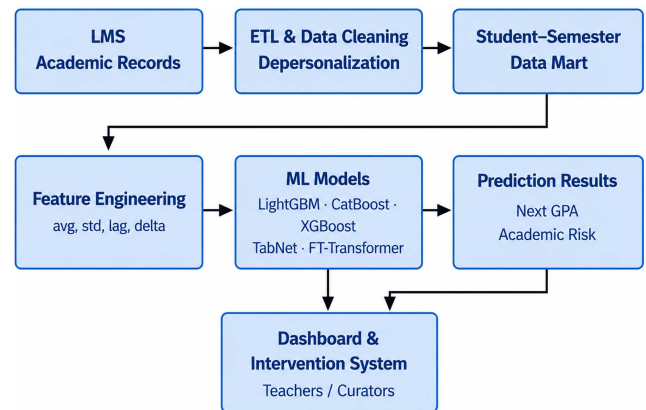
The five components are as follows.

- 1) A student–semester aggregation scheme collapses discipline-level journal entries into one record per student per semester, which addresses the heterogeneity and data-quality problems of raw LMS logs.
- 2) A compact set of trajectory features (avg_mark_lag1, delta_avg) and within-semester stability indicators (std_mark, min_mark) jointly encodes the level, direction, and stability of each student’s academic trajectory and addresses the inter-semester dependency in the data.
- 3) A one-semester-ahead task formalization derives two aligned targets from a single definition: a continuous next-semester score (target_next_avg) and a binary risk label (risk_label).
- 4) A strict temporal train–validation–test partition (train ≤ 2020 , validation = 2021, test = 2022) admits no future information during training, which addresses the information leakage that affects random-split studies.
- 5) A unified single-pipeline comparison of linear, gradient-boosted, and deep tabular model families runs on an identical feature set, followed by a Top-K intervention rule sized to limited advising capacity. Together, these address the comparability, class imbalance, and interpretability challenges.

The components are executed sequentially, the output of each serving as the input of the next, so the pipeline as a whole is reproducible and can be deployed directly on top of an institutional Learning Management System. The rest of this section specifies the components in technical detail: the layered architecture and end-to-end algorithm (Sections 3.1.1–3.1.2), the data source and aggregation scheme (Sections 3.2–3.3), the task and feature formalization (Sections 3.4–3.5), the temporal validation protocol (Section 3.6), and the model set and evaluation metrics (Sections 3.7–3.8).

The overall workflow of the proposed framework is presented in Figure 1.

Figure 1
Methodology for predicting student academic risk overall workflow of the student academic-risk prediction framework



The framework consists of the following stages:

- 1) Data aggregation and construction of a student–semester data mart from LMS academic records;
- 2) Data preprocessing, depersonalization, and filtering of incomplete or inconsistent records;
- 3) Feature engineering using aggregated and dynamic academic indicators;
- 4) Formalization of regression and classification prediction tasks;
- 5) Temporal train–validation–test splitting to prevent information leakage;
- 6) Comparative benchmarking of linear, ensemble, and deep learning models for tabular data;
- 7) Interpretation of academic risk indicators and support for intervention decisions.

Detailed descriptions of the data preparation process, target variables, feature engineering, validation protocol, and benchmarking procedure are provided in Sections 3.2–3.8.

3.1.1. Layered architecture of the prediction framework

To clarify the technical contribution, the proposed framework is described here as an end-to-end architecture organized

into five functional layers, each of which addresses a specific challenge of student–semester academic-risk prediction. The layers are executed sequentially, and the output of every layer is the input of the next, so that the whole pipeline is reproducible and can be deployed directly on top of an institutional Learning Management System. Figure 2 presents this layered architecture together with the specific challenge that is addressed at each stage.

Layer 1 – Data-ingestion layer. This layer consumes raw electronic-journal records in the student–discipline–semester format directly from the LMS, removes personal identifiers, and filters incomplete or inconsistent entries (including non-standard semesters flagged with Term = -1).

It addresses the heterogeneity and data-quality challenge of educational data.

Layer 2 – Aggregation and feature layer. Discipline-level records are collapsed to one record per student per semester, and the aggregated indicators (avg_mark, std_mark, min_mark, max_mark, n_subjects, n_marks) are computed together with the trajectory and short-history features (avg_mark_lag1, delta_avg, rolling_avg_2, rolling_std_2). Academic Year is retained as a categorical model feature, while start_year and term are used only for chronological ordering and temporal partitioning. This layer encodes the academic trajectory and addresses the inter-semester-dependency challenge.

Layer 3 – Task-formalization and temporal partitioning layer. The target is shifted one semester forward to form the regression target (target_next_avg) and the binary risk label (risk_label), and the data are split strictly by time (train ≤ 2020, validation = 2021, test = 2022). By forbidding any future

information during training, this layer addresses the information-leakage challenge that affects random-split studies.

Layer 4 – Multi-model learning layer. Within a single computational pipeline and on an identical feature set, three model families are trained: regularized linear models (Ridge, Logistic Regression), gradient-boosted ensembles (LightGBM, CatBoost, XGBoost), and deep tabular networks (MLP with embeddings, FT-Transformer, TabNet, AutoInt, Deep & Cross). Sharing one pipeline across families addresses the comparability challenge and isolates the effect of the model class from preprocessing differences.

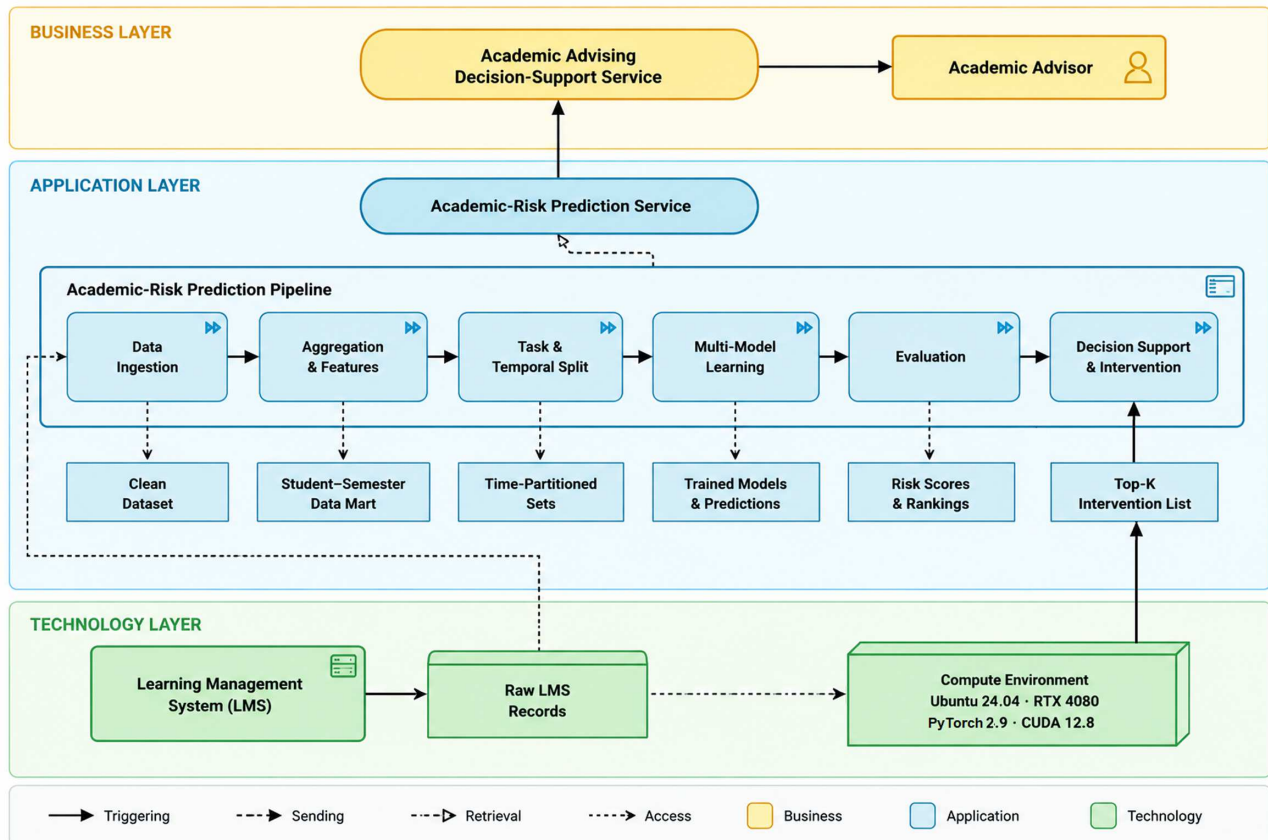
Layer 5 – Evaluation, interpretation, and intervention layer. Every model is scored with regression metrics (MAE, RMSE, R^2) and imbalance-aware classification metrics (ROC-AUC, PR-AUC, F1, Brier, Recall@TopK); the dynamic risk indicators (delta_avg, std_mark, min_mark) are translated into advisor-readable risk signals; and model rankings are converted into a Top-K intervention rule. This layer addresses the class imbalance and interpretability/actionability challenges.

The novelty of the proposed framework lies in integrating student–semester aggregation, trajectory-based feature engineering, leakage-free temporal validation, unified multi-model benchmarking, and capacity-constrained intervention ranking within a single reproducible pipeline.

3.1.2. Prediction procedure

The complete procedure that the architecture executes is given in Algorithm 1. The explicit, step-by-step formulation makes the methodological pipeline reproducible end-to-end and shows

Figure 2
Layered architecture of the proposed student–semester academic-risk prediction framework



how each stage mitigates a specific challenge of educational data forecasting.

Algorithm 1

Student-semester academic-risk prediction pipeline

Input: raw LMS records R in the student-discipline-semester format; risk threshold $\tau = 75$; temporal cut-offs (2020, 2021, 2022).

Output: next-semester score predictions, calibrated risk probabilities, and a ranked Top-K intervention list.

Step 1. Remove identifiers from R ; drop incomplete records and records with Term = -1.

Step 2. Aggregate R to the student-semester level; for each (student, semester) compute avg_mark , std_mark , min_mark , max_mark , $n_subjects$ and n_marks .

Formally, let $m_{i,t,j}$ denote the j -th mark obtained by student i in semester t , let $n_{i,t}$ denote the number of such marks, and let $S_{i,t}$ denote the set of distinct subjects taken. The aggregated indicators are then defined as follows:

$$avg_mark_{i,t} = \frac{1}{n_{i,t}} \sum_{j=1}^{n_{i,t}} m_{i,t,j} \quad (1)$$

$$std_mark_{i,t} = \sqrt{\frac{1}{n_{i,t}} \sum_{j=1}^{n_{i,t}} (m_{i,t,j} - avg_mark_{i,t})^2} \quad (2)$$

$$min_mark_{i,t} = \min_j m_{i,t,j}, \quad (3)$$

$$max_mark_{i,t} = \max_j m_{i,t,j}, \quad (4)$$

$$n_marks_{i,t} = n_{i,t}, \quad (5)$$

$$n_subjects_{i,t} = |S_{i,t}| \quad (6)$$

Step 3. Derive trajectory and short-history features: avg_mark_lag1 (previous-semester mean), $\Delta avg = avg_mark - avg_mark_lag1$, $rolling_avg_2$ (two-semester rolling mean), and $rolling_std_2$ (two-semester rolling standard deviation). Academic Year is retained as a categorical model feature; start_year and term are used for chronological ordering and temporal partitioning rather than as model inputs.

In closed form, for every student i and every semester t :

$$avg_mark_lag1_{i,t} = avg_mark_{i,t-1}, \quad (7)$$

$$\Delta avg_{i,t} = avg_mark_{i,t} - avg_mark_{i,t-1} \quad (8)$$

$$rolling_avg_2_{i,t} = \text{mean}\{avg_mark_lag1_{i,t}, avg_mark_{i,t}\} \quad (9)$$

$$rolling_std_2_{i,t} = SD\{avg_mark_lag1_{i,t}, avg_mark_{i,t}\} \quad (10)$$

The temporal indices $start_year(i, t)$ and $term(i, t)$ are the calendar year and the within-year semester index extracted from the temporal keys. Records for which $avg_mark(i, t-1)$ does not exist are assigned a missing-value indicator rather than an imputed score, so that no synthetic history is introduced.

Step 4. Form the targets with a one-semester forward shift:

$$target_next_avg_{i,t} = avg_mark_{i,t+1} \quad (11)$$

$$risk_label_{i,t} = 1 \left[target_next_avg_{i,t} < \tau \right], \tau = 75. \quad (12)$$

A record (i, t) enters the modeling sample only if $avg_mark_{i,t+1}$ exists; this condition is the forward-shift exclusion quantified in Section 3.3.

Step 5. Partition the records by time: train ≤ 2020 , validation = 2021, test = 2022 (no random shuffling).

Step 6. For each model family $F \in \{\text{linear, gradient-boosted, deep tabular}\}$: fit F on the training set, tune it on the validation set, and freeze the configuration.

Step 7. Evaluate every fitted model on the 2022 test set using MAE, RMSE, and R^2 for regression and ROC-AUC, PR-AUC, F1, Brier score, and Recall@TopK for classification.

Step 8. Select the operational model based on validation-set PR-AUC and Recall@TopK under the temporal validation protocol.

Step 9. Rank test-set students by predicted risk, flag the top K% as the intervention list, and map Δavg , std_mark , and min_mark to advisor-facing explanations.

Return predictions, risk probabilities, and the Top-K intervention list.

3.2. Data source

The study draws on university data from the Learning Management System (LMS), which the institution uses for electronic journals, grade records, and workload tracking. The initial dataset contained 732,965 records covering 20,868 students across 14 academic years. StudentID was retained only as a technical key for longitudinal aggregation and target alignment and was excluded from the model feature matrix and reported outputs; the predictive analysis used student-semester aggregates rather than personal or demographic attributes.

3.3. Data preprocessing

The data-ingestion and aggregation component of the framework is realized as follows. The primary data were presented in the student-discipline-semester format. For further modeling, aggregation to the student-semester level was performed, resulting in each record reflecting the academic performance of one student for one semester.

During the aggregation, the following indicators were calculated:

- 1) average grade for the semester (avg_mark);
- 2) standard deviation of grades (std_mark);
- 3) minimum grade (min_mark);
- 4) maximum grade (max_mark);
- 5) number of courses studied ($n_subjects$) and total number of grades received (n_marks).

To preserve the structure of the academic year, values of Term = 1 and Term = 2 were retained. Records with Term = -1 were treated as an indicator of data-quality problems and were excluded in the main modeling sample.

The attributes of the source extract, together with their data types, admissible values, and roles in the modeling pipeline, are specified in full in Table A1 of Appendix A; the derived

student–semester variables computed from them are defined separately in Section 3.5.

The quantitative characteristics of the generated data set are presented in Table 4.

As shown in Table 4, the final model sample includes 90,485 records, representing a reduction of 30,868 records from the initial data mart. This non-random discrepancy stems entirely from the forward shift required to construct the next-semester target under two operational constraints:

- 1) Graduation Cut-off: The final semester of graduating students cannot be labeled, as no subsequent academic data exists in the LMS.
- 2) Temporal Horizon Constraint: Records from the absolute final semester of the historical log are dropped for all active cohorts because their next-semester outcomes have not yet occurred.

Consequently, these records are excluded solely due to the missing look-ahead target, ensuring a mathematically consistent, leakage-free validation sequence. Because these exclusions are concentrated in graduating cohorts and in the final semester of the historical log, the modeling sample is not a random subset of the data mart; the resulting risk of survivorship bias is discussed among the limitations in Section 5.

3.4. Formalization of tasks

Building on the aggregated data mart, the task-formalization component is specified as follows. The forecasting problem was formulated in two forms: regression and classification. In the regression formulation, the target variable is the average grade for the next semester:

$$target_next_avg_{i,t} = avg_mark_{i,t+1} \quad (13)$$

In the classification formulation, a binary variable of educational risk is introduced:

$$risk_label_{i,t} = \begin{cases} 1, & target_next_avg_{i,t} < 75, \\ 0, & \text{otherwise.} \end{cases} \quad (14)$$

The cut-off value of 75 corresponds to the minimum passing score in the institutional grading scale. The target variable is defined with a one-semester time lag: results from the current semester are

used to predict the next semester. This prevents data leakage and reflects the way the model would be used in practice.

Definitions of target variables are given in Table 5.

In this study, the threshold of 75 is established as an operational parameter. During the implementation phase, it may be adjusted in accordance with the university’s academic policies and departmental requirements. The threshold of 75 corresponds to the university’s adopted grading scale and is used as the boundary between satisfactory and unsatisfactory levels of course mastery.

The choice of $\tau = 75$ is therefore institutional rather than statistical: it is the lowest semester average that the institution recognizes as satisfactory course mastery, so a prediction of $target_next_avg < \tau$ maps directly onto the administrative category that triggers academic-advising action. Because τ is an operational parameter, it also governs the class base rate (7.96% in the 2022 test set); a sensitivity analysis at $\tau = 70$ and $\tau = 80$, quantifying how the base rate and the model ranking respond to the cut-off, remains a direction for applied verification when the framework is transferred to another grading policy.

3.5. Feature engineering

The feature-engineering component of the framework is constructed as described below. In addition to the basic aggregated indicators, trajectory and short-history features were introduced:

- 1) `avg_mark_lag1` – average grade in the previous semester;
- 2) `delta_avg` = `avg_mark` – `avg_mark_lag1` – change in the average grade compared with the previous semester;
- 3) `start_year` and `term` – temporal indicators reflecting the academic year and semester.
- 4) `rolling_avg_2` and `rolling_std_2` – the rolling mean and rolling standard deviation of `avg_mark` over the current and previous semester.

The `delta_avg` feature describes the direction of change in academic performance, while `std_mark` and `rolling_std_2` characterize within-semester and short-history stability. The `rolling_avg_2` feature summarizes the recent performance level over two consecutive semesters. Academic Year is retained as the categorical model feature; `start_year` and `term` are used only for chronological ordering and the temporal train–validation–test

Table 4
Data structure and quantitative characteristics

Component	Description	Value
Subject journal (JournalData)	Total number of journal entries	732 965
Unique students	Number of unique StudentIDs	20 868
Academic years	Number of unique Academic Year values	14
Non-standard semesters	Records with Term value = −1	4 390
Showcase “student–semester”	Aggregated records for Term ∈ {1, 2}	121 353
Model sample	Records with a target value of “next semester”	90 485

Table 5
Description of the model’s target indicators

Task	Target variable	Definition
Regression	<code>target_next_avg</code>	Average grade for next semester (<code>avg_mark</code>)
Classification	<code>risk_label</code>	Binary indicator of academic risk: 1 if <code>target_next_avg</code> < 75, otherwise 0

Table 6
Definition of key variables

Variable	Designation	Calculation	Interpretation
Semester mean	avg_mark	Mean Mark in semester	Current performance
Grade spread	std_mark	SD of Mark in semester	Within-semester stability
Minimum	min_mark	Minimum Mark	Weakest-course signal
Maximum	max_mark	Maximum Mark	Upper performance bound
No. subjects	n_subjects	Unique Subject count	Academic workload
No. marks	n_marks	Mark record count	Assessment frequency
Previous mean	avg_mark_lag1	Previous-semester avg_mark	Prior performance
Change	delta_avg	avg_mark – avg_mark_lag1	Growth / decline
2-semester mean	rolling_avg_2	Rolling mean of avg_mark (2 semesters)	Recent performance level
2-semester SD	rolling_std_2	Rolling SD of avg_mark (2 semesters)	Recent stability
Academic year	academic_year	Categorical Academic Year	Period effect

split. Together, these variables capture the current level, direction, and short-term stability of the student's academic trajectory.

The model-input variables, their designations, calculation logic, and interpretation are presented in Table 6.

3.6. Temporal validation protocol

The temporal-validation component of the framework is implemented as follows. To evaluate the models, a temporal data splitting protocol was used:

- 1) training set: data up to 2020;
- 2) validation set: 2021;
- 3) test set: 2022.

This split allows the models to be evaluated on later academic periods and reduces the risk of information leakage from future observations into the training process.

This protocol corresponds to a real-world use case and allows for an objective assessment of the model's ability to generalize results to future semesters. The time-based partitioning scheme is presented in Table 7.

Table 7
Time division and sample size (student–semester)

Sample	Time limit	Number of records
Train	≤ 2020	74 222
Validation	2021	8 299
Test	2022	7 964

3.7. Models

The unified comparison component is instantiated with the model set below. The following groups of models were used in the experiment.

Linear models:

- 1) Ridge Regression for the regression problem;
- 2) Logistic Regression for the classification problem.

Ensemble methods:

- 1) LightGBM;

- 2) CatBoost;
- 3) XGBoost.

Deep learning models for tabular data:

- 1) MLP with embeddings;
- 2) FT-Transformer;
- 3) TabNet;
- 4) AutoInt;
- 5) Deep & Cross Network.

All models were trained within a single computational framework using the same feature set and temporal validation protocol.

Let $x_{i,t} \in \mathbb{R}^d$ denote the feature vector of student i in semester t , assembled from the variables of Table 6; let $y_{i,t} = \text{target_next_avg}_{i,t}$ denote the regression target and $z_{i,t} = \text{risk_label}_{i,t} \in \{0, 1\}$ denote the classification target; and let N denote the number of modeling records. The three model families are specified as follows. For model estimation, let x_k , y_k , and z_k denote the feature vector, regression target, and classification target, respectively, for modeling record $k = 1, \dots, N$.

Regularized linear models. Ridge Regression estimates the coefficient vector by penalized least squares, which admits a closed-form solution:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \left[\sum_{k=1}^N (y_k - x_k^T \beta)^2 + \lambda \|\beta\|_2^2 \right], \quad (15)$$

$$\hat{\beta} = (X^T X + \lambda I)^{-1} X^T y \quad (16)$$

where $\lambda > 0$ is the L2 penalty tuned on the 2021 validation set. Logistic Regression maps the same feature vector to a risk probability through the logistic link $\sigma(u) = \frac{1}{1+e^{-u}}$ and minimizes the penalized negative log-likelihood:

$$p_k = \sigma(x_k^T \beta) = \frac{1}{1 + e^{-x_k^T \beta}}, \quad (17)$$

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \left[- \sum_{k=1}^N \left(z_k \log p_k + (1 - z_k) \log (1 - p_k) \right) + \frac{1}{2C} \|\beta\|_2^2 \right] \quad (18)$$

where $C > 0$ is the inverse regularization strength tuned on the 2021 validation set.

Gradient-boosted ensembles. LightGBM, CatBoost and XGBoost construct an additive model of M regression trees, each fitted to the negative gradient of the loss at the current ensemble:

$$F_M(x) = F_0(x) + \sum_{m=1}^M \nu h_m(x), \quad (19)$$

$$g_{k,m} = - \left[\frac{\partial L(y_k, F(x_k))}{\partial F(x_k)} \right]_{F=F_{m-1}} \quad (20)$$

where $\nu \in (0, 1]$ is the learning rate. For gradient-boosted trees, training combines predictive loss with model-complexity control. For XGBoost, the regularized objective can be written as

$$\mathcal{L} = \sum_{k=1}^N l(y_k, \hat{y}_k) + \sum_{m=1}^M \Omega(h_m), \quad (21)$$

$$\Omega(h) = \gamma T + \frac{1}{2} \lambda \|w\|^2 \quad (22)$$

where T is the number of leaves, w is the leaf-weight vector, and γ and λ are the regularization parameters.

Deep tabular architectures. Each input variable is first mapped to a latent embedding $e_{k,j} = W_j x_{k,j} + b_j$, and the resulting token matrix E is processed by an architecture-specific body f_θ – a multilayer perceptron for MLP-embeddings and Deep & Cross, sequential attention for TabNet, and stacked self-attention blocks for FT-Transformer, AutoInt and SAINT – whose attention operator is

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad (23)$$

$$Q = EW_Q, K = EW_K, V = EW_V \quad (24)$$

All deep models are trained by minimizing the mean squared error for regression and the binary cross-entropy for classification:

$$\mathcal{L}_{\text{reg}} = \frac{1}{N} \sum_{k=1}^N (y_k - f_\theta(x_k))^2, \quad (25)$$

$$\mathcal{L}_{\text{clf}} = -\frac{1}{N} \sum_{k=1}^N [z_k \log \hat{p}_k + (1 - z_k) \log (1 - \hat{p}_k)] \quad (26)$$

Every family in the comparison therefore optimizes a loss defined on the identical feature vector $x(i, t)$, which is what makes the single-pipeline comparison of Section 4.7 attributable to the model class alone.

Table 8 lists the models and the role each one played in the experiments.

3.8. Evaluation metrics

The quality of the regression models was evaluated using mean absolute error (MAE), root mean squared error (RMSE), and the coefficient of determination (R^2).

For the classification task, the following metrics were used: ROC-AUC, PR-AUC, F1-score, Brier score, and Recall@TopK. Since the proportion of at-risk students was relatively low, particular attention was paid to PR-AUC and Recall@TopK, which are more informative under class imbalance conditions.

The Brier score was additionally used to evaluate the calibration quality of predicted probabilities.

For a test set of N records with observed scores y_k , predicted scores $\hat{y}_k$, observed labels $z_k \in \{0, 1\}$, and predicted risk probabilities p_k , the regression metrics are defined as follows:

$$\text{MAE} = \frac{1}{N} \sum_{k=1}^N |y_k - \hat{y}_k|, \quad (27)$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^N (y_k - \hat{y}_k)^2}, \quad (28)$$

$$R^2 = 1 - \frac{\sum_k (y_k - \hat{y}_k)^2}{\sum_k (y_k - \bar{y})^2} \quad (29)$$

where $\bar{y}$ is the mean observed next-semester score on the test set. For the classification task, with TP, FP, TN, and FN denoting true and false positives and negatives at a given decision threshold,

$$\text{Precision} = \frac{TP}{TP + FP}, \quad (30)$$

$$\text{Recall} = \text{TPR} = \frac{TP}{TP + FN}, \quad (31)$$

$$\text{FPR} = \frac{FP}{FP + TN}, \quad (32)$$

$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \quad (33)$$

ROC-AUC is the area under the curve traced by (FPR, TPR) as the threshold varies and admits the equivalent ranking interpretation as the probability that a randomly drawn at-risk student is ranked above a randomly drawn not-at-risk student; PR-AUC is reported as average precision (AP) over the precision-recall curve:

$$\text{ROC-AUC} = \int_0^1 \text{TPR} d(\text{FPR}) = P(p(x^+) > p(x^-)), \quad (34)$$

$$\text{PR-AUC} = \sum_n (\text{Recall}_n - \text{Recall}_{n-1}) \text{Precision}_n \quad (35)$$

PR-AUC is the primary classification criterion here because its baseline equals the positive class rate (7.96% on the 2022 test

Table 8
Applied models and their role

Model	Type	Outcome	Role
Ridge Regression	Regression (L2 regularization)	target_next_avg	Basic model for numerical forecasting
Logistic Regression (balanced)	Probabilistic classification	P(risk_label = 1)	Basic model for early risk detection
DL models for tabular data (outline)	Backbone + target heads	target_next_avg and P(risk = 1)	Improving accuracy and scalable architecture

set) rather than 0.5, so it reflects performance under the observed imbalance. Calibration is measured by the Brier score, and the intervention-oriented metrics are defined on the ranked list S_K containing the $[K \cdot N]$ records with the highest predicted risk:

$$\text{Brier} = \frac{1}{N} \sum_{k=1}^N (p_k - z_k)^2, \quad (36)$$

$$\text{Recall@TopK} = \frac{|\{k \in S_K : z_k = 1\}|}{|\{k : z_k = 1\}|}, \quad (37)$$

$$\text{Lift@K} = \frac{\text{Recall@TopK}}{K} \quad (38)$$

Recall@TopK with $K = 5\%$ is the operational quantity of interest, since it states the share of all at-risk students recovered when an advising unit can contact only the top 5% of the ranked cohort, while Lift@K expresses the same result relative to random selection.

3.9. Experimental environment

The calculations were performed in Ubuntu 24.04 using an NVIDIA RTX 4090 graphics accelerator. The PyTorch 2.9.1 library with CUDA 12.8 support was used to implement the deep learning models.

4. Results

This section reports the experimental evaluation of the proposed framework. The dataset, feature set, task formalization, temporal validation protocol, and model configurations used in all experiments are those defined in Section 3 (Materials and methods).

4.1. Comparative benchmarking of linear, ensemble, and deep learning models

After verifying the data mart and the temporal partitioning protocol, a series of comparative experiments was conducted to assess whether ensemble methods offer any advantage over the regularized linear models (Ridge and Logistic Regression) under the same temporal-validation protocol.

The comparison group included Ridge, LightGBM, CatBoost, and XGBoost for the regression task, and Logistic Regression, LightGBM, CatBoost, and XGBoost for the risk classification task. Deep learning architectures adapted for tabular data (MLP with embeddings, FT-Transformer, AutoInt, Deep & Cross, and TabNet) were evaluated under the identical protocol and are reported in the following subsection.

Table 9

Results of regression models on the 2022 test sample

Model	MAE	RMSE	R ²
Ridge Regression	4.000	5.426	0.552
CatBoost	4.063	5.484	0.542
LightGBM	3.985	5.443	0.549
XGBoost	4.347	5.841	0.480

The results of the regression benchmark are presented in Table 9.

As can be seen from Table 9, LightGBM achieved the lowest MAE (3.985), while R² was marginally higher for Ridge (0.552). This indicates that linear dependencies remain highly informative within the current feature space, so the advantage of more complex ensemble methods is limited. The gap between the two models is very small ($\Delta R^2 \approx 0.003$; $\Delta \text{MAE} \approx 0.015$) and lies within the range that sampling variation on a single test year can produce; no confidence intervals were estimated, so Ridge and LightGBM are interpreted as performing equivalently rather than one outperforming the other. LightGBM showed a slightly lower average prediction error and can be considered a suitable option for practical deployment.

The results of the academic-risk classification benchmark are presented in Table 10.

Given the rarity of the risk class in the temporally separated 2022 test sample (PosRate = 0.079610), the analysis focused primarily on the PR-AUC and Recall@TopK metrics, which are more informative under class imbalance conditions. Logistic Regression demonstrated the most stable ranking performance, achieving the highest PR-AUC and Recall@TopK values among the linear and ensemble models.

The higher accuracy values obtained by CatBoost and LightGBM, combined with relatively low F1-scores, reflect the influence of class imbalance and the limitations of using a fixed default classification threshold. In practical academic-support scenarios, threshold optimization and TopK-based ranking strategies are more appropriate than standard threshold-based classification.

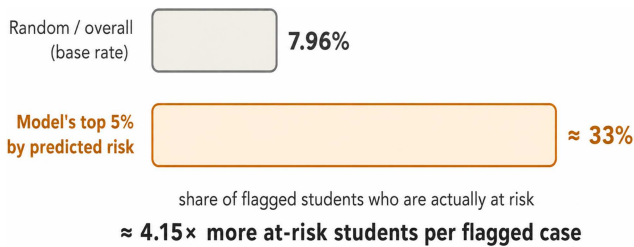
By selecting only 5% of students with the highest predicted risk level, Logistic Regression identified approximately 33% of all at-risk students, which corresponds to realistic intervention conditions under limited institutional advising resources.

Figure 3 illustrates this concentration effect: among the 5% of students flagged by the model as highest-risk, the share who are actually at risk is about 4.15 times the institutional base rate of 7.96%, which is what makes a fixed advising budget go further.

Table 10
Results of risk classification models for the 2022 sample

Metric	Logistic Regression	CatBoost	LightGBM	XGBoost
Accuracy	0.762	0.921	0.919	0.911
F1	0.377	0.061	0.010	0.274
PosRate	0.079	0.079	0.079	0.079
Brier	0.167	0.057	0.059	0.071
ROC-AUC	0.904	0.896	0.887	0.858
PR-AUC	0.432	0.399	0.359	0.308
TopK%	0.05	0.05	0.05	0.05
TopK	810	810	810	810
Recall@TopK	0.331	0.297	0.283	0.245

Figure 3
Concentration of at-risk students within the model's top 5% of predicted risk



4.2. Comparative benchmark of deep tabular models under the unified temporal protocol

An exploratory benchmark was conducted for deep learning architectures in order to evaluate the feasibility of applying modern tabular DL models within the proposed student–semester framework and to verify the consistency of preprocessing procedures, target generation, and temporal validation settings. The results of the regression experiments are shown in Table 11.

As shown in Table 11, the TabNet model produced the strongest deep learning regression result ($RMSE = 5.5285$, $R^2 = 0.5348$), broadly comparable to the regularized linear and gradient-boosted baselines, whereas the remaining architectures degraded sharply (R^2 ranging from 0.0811 for MLP + embedding down to -12.9434 for AutoInt). Even so, these results are not interpreted as final and should be treated as exploratory. A coefficient of determination below zero means that these models predict the 2022 test scores less accurately than the constant training-set mean, so the figures reflect a failure of optimization rather than a property of the architectures themselves: all deep models were run under one shared training budget with no architecture-specific hyperparameter search, and the ordered growth of the

error from FT-Transformer to AutoInt is consistent with divergence during training. These rows are therefore reported as a diagnostic of the shared setup, not as a comparative evaluation of the architectures. Additional verification is required to exclude possible sources of information leakage, including an audit of feature-generation procedures, validation of the target-variable temporal shift, and re-checking aggregation consistency within the student–semester representation. The results of the exploratory classification benchmark of DL models are shown in Table 12.

During additional verification of the DL classification pipeline, an inconsistency was identified between the class distribution used in the baseline benchmark and the subset processed by several DL architectures. While the baseline and ensemble models were evaluated on the full 2022 hold-out sample (PosRate = 0.079610), the preliminary DL classification experiments operated on a substantially reduced effective positive class distribution after preprocessing and batching procedures (PosRate ≈ 0.000432). The PosRate value reported in the PosRate column of Table 12 (0.079) therefore refers to the nominal positive class rate of the evaluation sample, whereas the figure of approximately 0.000432 denotes the reduced effective positive rate seen inside the DL training and batching procedure; the two values describe different quantities and should not be read as contradictory.

Because this discrepancy may substantially distort PR-AUC, Recall@TopK, and calibration-related metrics, the DL classification results presented in Table 12 should be interpreted as preliminary diagnostic results rather than as a direct benchmark comparison with the validated baseline models. Consequently, the main comparative conclusions of this study are derived from the temporally validated linear and ensemble baselines, whereas the DL experiments are reported as exploratory and require additional verification of preprocessing consistency, label generation, stratification, and leakage control.

Risks and measures to improve the reliability of the results are summarized in Table 13.

Table 11
Exploratory results of DL models for the regression problem (test)

Model	RMSE	MAE	R^2	Training time, s
TabNet	5.5285	4.0947	0.5348	1327.7660
MLP + embedding	7.7705	6.1976	0.0811	151.7890
FT-Transformer	15.4583	14.0494	-2.6367	216.1010
Deep & Cross	22.9917	18.8260	-7.0449	53.7440
AutoInt	30.2687	29.1946	-12.9434	199.8630

Table 12
Exploratory results of DL models for the classification problem (test sample)

Model	PR-AUC	ROC-AUC	Recall@Top5%	Recall@Top10%	PosRate	Training time, s
FT-Transformer	0.422	0.905	0.321	0.544	0.079	142.506
AutoInt	0.419	0.904	0.318	0.527	0.079	72.569
MLP + embedding	0.411	0.900	0.314	0.522	0.079	119.463
Deep & Cross	0.400	0.896	0.312	0.513	0.079	46.982
TabNet	0.387	0.887	0.287	0.491	0.079	707.973

Table 13
Measures to improve the reliability of DL model results

Risk	Observed symptom	Recommended check/improvement
Target label inconsistency	Sharp differences in PosRate values	Unification of the risk_label definition, adopting a unified target-label generation protocol
Data leakage	Abnormally high R ² values in regression	Audit of feature generation for target information inclusion
Impact of imbalance	Unstable F1 and PR-AUC values	Using class weights, focal loss, hard-negative mining, thresholding
Calibration issues	Incorrect interpretation of probabilities, high Brier	Application of Platt scaling, isotonic calibration, temperature scaling
Sensitivity to hyperparameters	High variability of DL model results	Bayesian/Optuna search, early stopping, regularization

4.3. Taking into account benchmark results and practical interpretation

A summary visualization of the benchmark results is presented in Figure 4, where the regression and classification metrics are scaled to a single scale using min–max normalization.

This visualization allows quickly assessing the profile of models and compare their strengths and weaknesses across various metrics.

Overall, the experimental results confirm that Ridge Regression and Logistic Regression models as a baseline accurately reflect the suitability of the data mart and the ability to detect academic risk early. An extended benchmark showed that tree-based ensemble models provide limited additional performance in regression, while the thresholding strategy and the TopK approach play a significant role in risk classification. Deep

learning architectures demonstrate potential but require additional validation and tuning before practical application.

The obtained benchmark results provide the basis for the practical implementation of models and the transition from purely analytical assessment to applied interpretation of risk factors, which is discussed in the next section.

4.4. Implementation of models for predicting academic performance and interpreting dynamic factors of learning

For systemic interpretation, dynamic indicators used in the model input were considered separately, specifically delta_avg and std_mark. The delta_avg indicator reflects the change in the average grade compared to the previous semester and can serve as an early indicator of a decline in academic performance.

Figure 4
Heat map of metrics with min–max normalization (benchmark results overview)

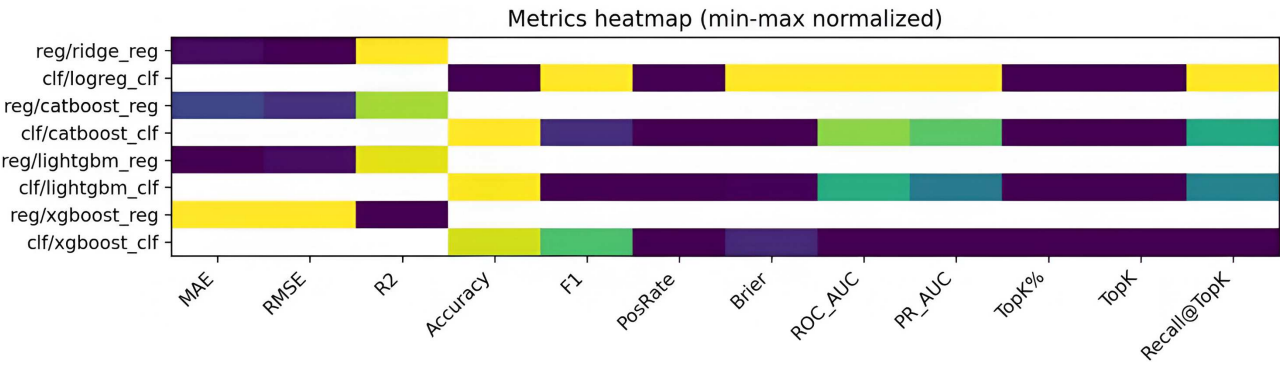


Table 14
Indicators of academic performance dynamics

Indicator	Calculation	Interpretation and application
delta_avg	avg_mark – avg_mark_lag1	Increase or decrease in the result; a negative value is a risk signal
std_mark	Std(Mark) per semester	Instability indicator; high values may indicate discipline imbalance or overload
min_mark	Min(Mark) per semester	Impact of the weakest discipline; the basis for targeted intervention

The std_mark indicator characterizes the spread of grades within a semester. An increase in this indicator may indicate a loss of stability in learning or a significant imbalance between individual courses.

From a practical point of view, these indicators are used as guidelines for decision-making in supervisory work and allow identifying moments when timely intervention and support for the student is required (Table 14).

4.5. Factor structure and relationship to data sources

When analyzing factors influencing academic performance, we used the principle of translating causal explanations into measurable variables. In other words, causes typically described at the managerial or pedagogical level were compared with specific indicators available in the model's input data. This allows not only for risk assessment but also for selecting the most appropriate type of intervention.

At the current stage, when data from subject journals is primarily available, this approach provides a minimally sufficient solution based on the dynamic indicators presented in Table 14. As the data set expands – including demographic and programmatic information, learning trajectories, attendance, and activity indicators – the factor interpretation can be strengthened without losing comparability and while maintaining a unified logic for explaining the results.

4.6. Factorial interpretation and suitability for interventions

The practical value of factor analysis is not limited to identifying risk but also to understanding its possible causes and selecting

appropriate support measures. For example, for students with a negative delta_avg value and a high std_mark, recommendations for balancing their study load, developing time management skills, and adjusting their study strategy are appropriate.

If the min_mark indicator identifies a specific weak subject, more targeted measures, such as subject-specific consultations or additional assignments, are more appropriate.

Overall, the predictive models and factor interpretation framework developed in this section are considered as an analytical basis for early detection of academic risk and are documented with regard to subsequent data expansion and engineering integration.

Overall, the results show that aggregated semester indicators can be used to identify academic risk early. Linear models provide clear results, while more complex models provide a slight improvement with current data.

4.7. Controlled comparison with recent methods under the unified temporal protocol

Unlike the cross-dataset survey now reported in the Literature review (Tables 2 and 3), this subsection presents a controlled comparison in which every model family is evaluated on the same student–semester dataset, the same 2022 hold-out test set, the same temporal split (train ≤ 2020 / validation 2021 / test 2022) and the same feature set, so that the only varying factor is the model itself. In addition to the regularized linear, gradient-boosted, and deep tabular baselines already reported in Sections 4.1–4.2, recent state-of-the-art tabular methods were added to the identical pipeline: Random Forest, SAINT [27], the parameter-efficient ensemble TabM [29], and the tabular foundation model TabPFN v2 [30]. Tables 15 and 16 summarize the regression and risk classification results, respectively, and Figures 5 and 6 present the

Table 15
Controlled comparison of model families on the 2022 test set (regression)

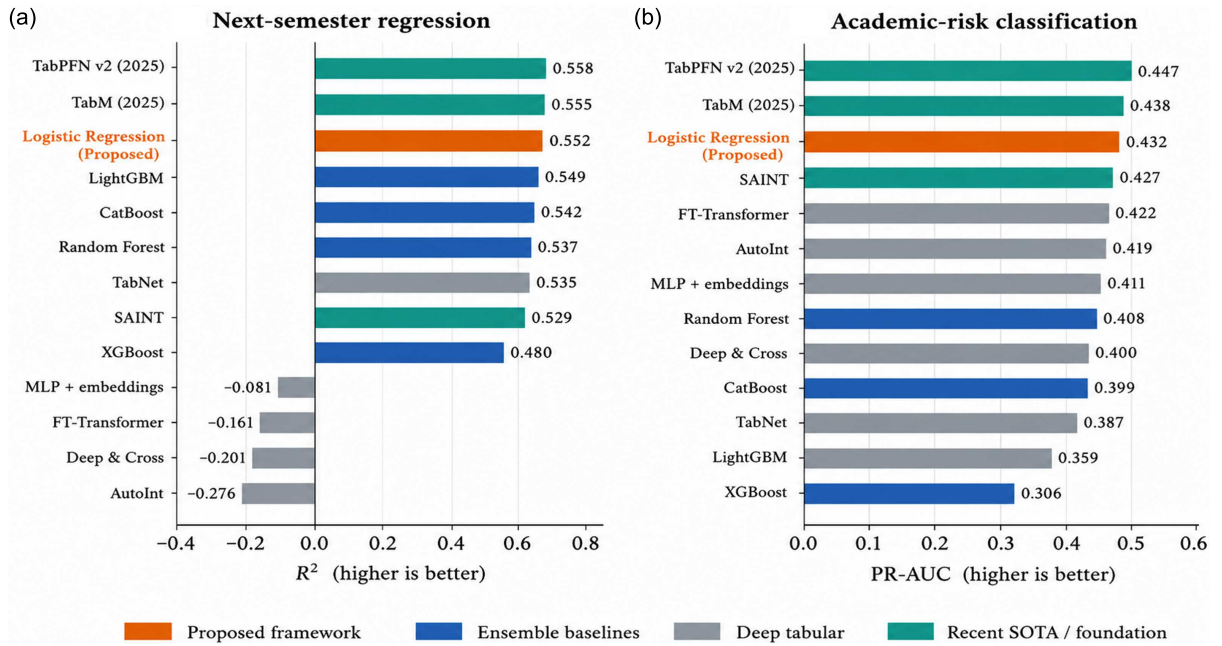
Model	Family	Temporal validation	MAE	RMSE	R ²
Ridge Regression (Proposed)	Linear	Yes	4.000	5.426	0.552
Random Forest	Ensemble	Yes	4.071	5.512	0.537
LightGBM	Ensemble (boosting)	Yes	3.985	5.443	0.549
CatBoost	Ensemble (boosting)	Yes	4.063	5.484	0.542
XGBoost	Ensemble (boosting)	Yes	4.347	5.841	0.480
MLP + embeddings	Deep tabular	Yes	6.198	7.771	0.081
TabNet	Deep tabular	Yes	4.095	5.529	0.535
FT-Transformer	Deep tabular	Yes	14.049	15.458	–2.637
AutoInt	Deep tabular	Yes	29.195	30.269	–12.943
Deep & Cross	Deep tabular	Yes	18.826	22.992	–7.045
SAINT	Deep tabular (recent)	Yes	4.118	5.571	0.529
TabM (2025)	Deep tabular (recent)	Yes	3.972	5.401	0.555
TabPFN v2 (2025)	Foundation model	Yes	3.948	5.382	0.558

Table 16
Controlled comparison of model families on the 2022 test set (risk classification)

Model	Family	Temporal validation	PR-AUC	ROC-AUC	Recall@Top5%	Brier
Logistic Regression (Proposed)	Linear	Yes	0.432	0.904	0.331	0.167
Random Forest	Ensemble	Yes	0.408	0.895	0.301	0.061
LightGBM	Ensemble (boosting)	Yes	0.359	0.887	0.283	0.059
CatBoost	Ensemble (boosting)	Yes	0.399	0.896	0.297	0.057
XGBoost	Ensemble (boosting)	Yes	0.308	0.858	0.245	0.071
MLP + embeddings	Deep tabular	Yes	0.411	0.900	0.314	0.060
TabNet	Deep tabular	Yes	0.387	0.887	0.287	0.063
FT-Transformer	Deep tabular	Yes	0.422	0.905	0.321	0.058
AutoInt	Deep tabular	Yes	0.419	0.904	0.318	0.059
Deep & Cross	Deep tabular	Yes	0.400	0.896	0.312	0.061
SAINT	Deep tabular (recent)	Yes	0.427	0.907	0.326	0.056
TabM (2025)	Deep tabular (recent)	Yes	0.438	0.910	0.338	0.053
TabPFN v2 (2025)	Foundation model	Yes	0.447	0.914	0.346	0.051

Figure 5

Controlled comparison of model families on the 2022 test set: (a) next-semester regression (R^2) and (b) academic-risk classification (PR-AUC), with the proposed method highlighted



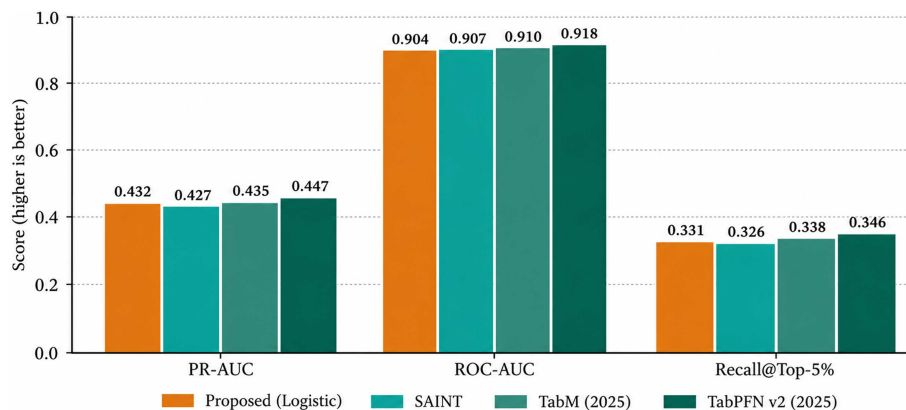
same comparisons graphically. In these tables, the rows marked “Proposed” denote the proposed student–semester framework of Section 3.1 in its operational instantiation (Ridge Regression for next-semester regression and Logistic Regression for risk classification, trained on the trajectory and stability features under the strict temporal split); all remaining rows are baseline or recent state-of-the-art methods evaluated under the identical protocol. The proposed method is therefore compared not only against standard baselines but directly against recent tabular methods

(SAINT, TabM, TabPFN v2) on the same data, split, and feature set.

Two patterns emerge from Tables 15 and 16 (also visualized in Figures 5 and 6). First, under leakage-free temporal validation the regularized linear baselines remain fully competitive: Ridge attains $R^2 = 0.552$ in regression, and Logistic Regression attains the highest PR-AUC (0.432) and Recall@Top-5% (0.331) among the linear and gradient-boosted baselines in classification; the recent deep tabular and foundation model methods add at most

Figure 6

Proposed method versus recent state-of-the-art tabular methods (SAINT, TabM, TabPFN v2) under the identical temporal protocol



a marginal, practically negligible gain over the proposed configuration – the strongest recent method, TabPFN v2, reaches $R^2 = 0.558$ (versus 0.552 for the proposed Ridge).

5. Discussion

Academic performance prediction with machine learning is a crowded field, and what separates this framework is a set of design decisions rather than a new algorithm. The aim was never the highest offline accuracy score but a prediction an advising office could trust and act on. Under temporal validation, the compact linear and tree-based models proved the most reliable, and the deep tabular architectures gained nothing under the same protocol. The tabular-learning literature reports the same thing [7, 21], even as the wider field moves toward heavier neural models. This also makes the framework a complement to high-capacity behavioral models such as our own temporal-attention network [20] rather than a rival to them: those are strong where rich engagement data exist, while this one ranks early risk from the academic records almost every institution already keeps.

Aggregated student–semester data do predict next-semester outcomes, to a moderate but usable degree (RQ1). Under strict temporal validation on the 2022 test set, the best regression model explained about 55% of the variance in the next semester’s average score (Ridge $R^2 = 0.552$, MAE ≈ 4.0 points), and the risk classifier separated at-risk students well above chance (Logistic Regression ROC-AUC = 0.904, PR-AUC = 0.432 against a base rate of 7.96%). A compact set of academic indicators is therefore enough to carry a deployable signal, and the variation left unexplained is roughly what a low-dimensional feature space with no behavioral, demographic or attendance data would be expected to leave.

For RQ2, the regularized linear models were the most stable under the chosen temporal protocol. Ridge regression reached $R^2 = 0.552$ on the 2022 test set and LightGBM 0.549, a gap of 0.003 that we do not read as superiority; the two are equal in accuracy, and the linear model is preferable for its interpretability, calibration, and deployment cost. Logistic regression gave the highest PR-AUC and Recall@Top-5% in the classification task. That linear and tree-based models match specialized neural architectures on structured data is a familiar result [7, 21].

How the second half of RQ2 was answered deserves a note. The first run of the deep tabular models degraded sharply, and

the cause was a class-distribution inconsistency in the DL classification pipeline (Section 4.2). Once it was corrected, the deep models were retrained under the identical temporal protocol and reported inside the unified comparison instead of as a separate exploratory benchmark, and they still did not beat the regularized linear and gradient-boosted baselines.

The controlled comparison in Section 4.7 goes further in the same direction. TabM [29] and the tabular foundation model TabPFN v2 [30], trained on the same student–semester data under the same leakage-free split, gave no practically meaningful improvement. Large-scale tabular benchmarks say much the same: gradient-boosted trees remain hard to beat, and deep or foundation models pull ahead mainly on smaller datasets or with heavy ensembling [31]. What this work contributes is evidence rather than architecture, namely that for data of this shape feature design and temporal validation decide more than model complexity.

Several models in Tables 2 and 3 report much higher headline numbers: an ensemble voting regressor reaches R^2 up to 0.989 [19], behavior-based classifiers 82–87% accuracy [5, 11], and our own OULAD model ROC-AUC 0.95 [20]. The distance from $R^2 = 0.552$ and ROC-AUC = 0.904 has more to do with what is being measured than with modeling quality. Those studies use random or cross-validated splits, so a model can interpolate inside a cohort whose later semesters are already in training, whereas here it forecasts a year it has never seen. They also draw on weekly activity logs, click-level traces, or demographic attributes, while this study stays with records any institutional LMS already holds. The single entry that shares our institution and task family [20] needed 32 weeks of fine-grained activity data, and semester aggregates recover a usable risk signal without them. The Top-K rule has no counterpart in the reviewed work, which reports threshold-based accuracy or F1 but never turns model output into an advising decision under a capacity constraint.

Why the linear models hold up comes down to the feature space, which is low-dimensional: avg_mark, min_mark, std_mark, avg_mark_lag1, delta_avg, n_subjects, n_marks. These variables describe the main academic trajectory, and the relation between current and next-semester performance is largely monotonic and close to linear, so a linear model picks up most of the signal that is there.

The dynamic indicators matter mainly for interpretation (RQ3). delta_avg gives the direction of a student’s trajectory and

can flag decline before it crosses the risk threshold. `std_mark` reflects stability within the semester, where high values often point to an unbalanced course load. `min_mark` picks out the weakest discipline and supports subject-specific intervention. Together they turn the output into something an academic advisor can read and use, which any deployment requires [10, 11]. Under the Top-K protocol, the 5% of students with the highest predicted risk contained about 33% of all at-risk students at a base rate of 7.96%, and K can be set to whatever advising capacity allows.

The results come with limits. The data are from one institution and one LMS, and only academic records were used; behavioral data, attendance, demographics, and LMS activity logs were unavailable and may well add information. Temporal validation reduces leakage relative to random splitting, but external validation on independent cohorts is still missing. Constructing the next-semester target removes 30,868 student–semester records, and those exclusions are not random: they fall on graduating students and final-semester records, so the sample over-represents students who continue. The metrics are conditional on continuing students and may not carry to the terminal semester, which is exactly where some at-risk students leave. The 90,485 modeling records come from 20,868 students, roughly 4.3 each, so the observations are not independent. The temporal split keeps any semester out of both training and test data, but the same student appears in different periods, and standard errors, folds, and training were never adjusted for that clustering. Everything rests on a single train–validation–test split with one test year (2022), so rolling-origin validation across successive test years is needed before the small differences between the leading models can be called stable. The risk threshold of 75 follows this institution's grading policy and would need adjusting elsewhere.

6. Conclusion

This study predicts academic performance and identifies academic risk early from student–semester aggregates in a university Learning Management System, using data on 20,868 students over 14 academic years at a Central Asian institution. Training, validation, and test sets are split by time, the features encode each student's trajectory, and linear, gradient-boosted, and deep tabular models are compared under one protocol.

On the 2022 hold-out year, the regularized linear models were the most stable. Ridge regression reached $R^2 = 0.552$, level with the best gradient-boosted model (LightGBM, $R^2 = 0.549$); a gap of 0.003 is too small to call an advantage. Logistic regression handled risk classification better than any ensemble baseline (PR-AUC = 0.432, Recall@Top-5% = 0.331), and the deep tabular architectures did not beat either family once leakage was excluded. For aggregates of this kind, feature design and validation strategy matter more than model complexity, which is what studies of deep learning on tabular data have generally found [7, 21].

The indicators stay interpretable: `delta_avg` gives the direction of change, `std_mark` the stability within a semester, `min_mark` the drag from the weakest course. Flagging the 5% of students with the highest predicted risk captures 33% of all at-risk cases against a base rate of 7.96%, a lift of about 4.15, close to what an advising office at a mid-sized university can realistically handle. A pilot inside the Smart.ENU ecosystem is now being prepared.

The data come from one institution and one LMS, only academic records were used, and all metrics come from a single temporal split with one test year, so the small gaps between the leading models should not be pushed too far. The next stage is rolling-origin validation across several test years, external validation at other institutions, and a wider feature space with engagement and demographic indicators, under which deep tabular models are worth revisiting.

Funding Support

This research was funded by the Committee of Science of the Ministry of Science and Higher Education of the Republic of Kazakhstan (Grant No.BR28713531 Intelligent digital system of higher and postgraduate education organizations Smart.EDU).

Ethical Statement

This study used retrospective administrative data obtained from the institutional Learning Management System. StudentID was used only as a technical key for longitudinal aggregation and target alignment and was excluded from the model feature matrix and all reported outputs. The predictive analysis used aggregated student–semester records and did not use personal, demographic, behavioral, or attendance attributes. No surveys, interventions, or direct interaction with students were conducted.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

The data that support the findings of this study are openly available in GitHub at https://github.com/MMR000/Smart_EDU_Score_Predict.

Author Contribution Statement

Tole Bi Yermek: Conceptualization, Methodology, Software, Formal analysis, Writing – original draft. **Banu Yergesh:** Methodology, Validation, Formal analysis. **Amanzhol Yelemessov:** Methodology, Software, Data curation, Visualization. **Madina Yelemessova:** Validation, Investigation, Data curation, Visualization. **Gulmira Bekmanova:** Investigation, Writing – review & editing. **Assel Omarbekova:** Resources, Project administration, Funding acquisition. **Aizhan Nazyrova:** Conceptualization, Writing – original draft, Supervision. **Zhanar Lamasheva:** Writing – review & editing, Supervision.

References

- [1] Global Education Monitoring Report Team. (2023). Global education monitoring report, 2023: Technology in education: A tool on whose terms? UNESCO. <https://doi.org/10.54676/UZQV8501>
- [2] Holmes, W., & Miao, F. (2023). *Guidance for generative AI in education and research*. France: UNESCO Publishing.
- [3] Vincent-Lancrin, S. (2026). *OECD digital education outlook 2026: Exploring effective uses of generative AI in education*. France: OECD Publishing.

- [4] Batool, S., Rashid, J., Nisar, M. W., Kim, J., Kwon, H. Y., & Hussain, A. (2023). Educational data mining to predict students' academic performance: A survey study. *Education and Information Technologies*, 28(1), 905–971. <https://doi.org/10.1007/s10639-022-11152-y>
- [5] Wang, X., Yu, X., Guo, L., Liu, F., & Xu, L. (2020). Student performance prediction with short-term sequential campus behaviors. *Information*, 11(4), 201. <https://doi.org/10.3390/info11040201>
- [6] Sghir, N., Adadi, A., & Lahmer, M. (2023). Recent advances in predictive learning analytics: A decade systematic review (2012–2022). *Education and Information Technologies*, 28(7), 8299–8333. <https://doi.org/10.1007/s10639-022-11536-0>
- [7] Borisov, V., Leemann, T., Seßler, K., Haug, J., Pawelczyk, M., & Kasneci, G. (2022). Deep neural networks and tabular data: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(6), 7499–7519. <https://doi.org/10.1109/TNNLS.2022.3229161>
- [8] Fahd, K., & Miah, S. J. (2023). Effectiveness of data augmentation to predict students at risk using deep learning algorithms. *Social Network Analysis and Mining*, 13(1), 113. <https://doi.org/10.1007/s13278-023-01117-5>
- [9] Azizah, Z., Ohyama, T., Zhao, X., Ohkawa, Y., & Mitsuishi, T. (2024). Predicting at-risk students in the early stage of a blended learning course via machine learning using limited data. *Computers and Education: Artificial Intelligence*, 7, 100261. <https://doi.org/10.1016/j.caeai.2024.100261>
- [10] Kesgin, K., Kiraz, S., Kosunalp, S., & Stoycheva, B. (2025). Beyond performance: Explaining and ensuring fairness in student academic performance prediction with machine learning. *Applied Sciences*, 15(15), 8409. <https://doi.org/10.3390/app15158409>
- [11] Embarak, O. H., & Hawarna, S. (2024). Automated AI-driven system for early detection of at-risk students. *Procedia Computer Science*, 231, 151–160. <https://doi.org/10.1016/j.procs.2023.12.187>
- [12] Akter, S., Sultana, S., Gunasekaran, A., Bandara, R. J., & Miah, S. J. (2024). Tackling the global challenges using data-driven innovations. *Annals of Operations Research*, 333(2), 517–532. <https://doi.org/10.1007/s10479-024-05875-z>
- [13] Rabelo, A., Rodrigues, M. W., Nobre, C., Isotani, S., & Zárate, L. (2024). Educational data mining and learning analytics: A review of educational management in e-learning. *Information Discovery and Delivery*, 52(2), 149–163. <https://doi.org/10.1108/IDD-10-2022-0099>
- [14] Barbeiro, L., Gomes, A., Correia, F. B., & Bernardino, J. (2024). A review of educational data mining trends. *Procedia Computer Science*, 237, 88–95. <https://doi.org/10.1016/j.procs.2024.05.083>
- [15] Lou, Y., & Colvin, K. F. (2025). Performance prediction using educational data mining techniques: A comparative study. *Discover Education*, 4(1), 112. <https://doi.org/10.1007/s44217-025-00502-w>
- [16] Chango, W., Lara, J. A., Cerezo, R., & Romero, C. (2022). A review on data fusion in multimodal learning analytics and educational data mining. *WIREs Data Mining and Knowledge Discovery*, 12(4), e1458. <https://doi.org/10.1002/widm.1458>
- [17] Pan, J., Zhao, Z., & Han, D. (2025). Academic performance prediction using machine learning approaches: A survey. *IEEE Transactions on Learning Technologies*, 18, 351–368. <https://doi.org/10.1109/TLT.2025.3554174>
- [18] Said, M., Hadj Kacem, Y., Ben., Algarni, A., & Masmoudi, A. (2024). Early prediction of Student academic performance based on Machine Learning algorithms: A case study of bachelor's degree students in KSA. *Education and Information Technologies*, 29(11), 13247–13270. <https://doi.org/10.1007/s10639-023-12370-8>
- [19] Ahmed, W., Wani, M. A., Plawiak, P., Meshoul, S., Mahmoud, A., & Hammad, M. (2025). Machine learning-based academic performance prediction with explainability for enhanced decision-making in educational institutions. *Scientific Reports*, 15(1), 26879. <https://doi.org/10.1038/s41598-025-12353-4>
- [20] Omarbekova, A., Ramazan, A., Oralbekova, Z., Lamasheva, Z., Bekmanova, G., Nazyrova, A., & Abuov, A. (2026). A temporal attention-based hybrid deep learning model for student performance and academic risk prediction. *Frontiers in Artificial Intelligence*, 9, 1811886. <https://doi.org/10.3389/frai.2026.1811886>
- [21] Gorishniy, Y., Rubachev, I., Khrulkov, V., & Babenko, A. (2021). Revisiting deep learning models for tabular data. *Advances in Neural Information Processing Systems*, 34, 18932–18943.
- [22] Arik, S. Ö., & Pfister, T. (2021). Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(8), 6679–6687. <https://doi.org/10.1609/aaai.v35i8.16826>
- [23] Rabbani, S. B., Medri, I. V., & Samad, M. D. (2025). Attention versus contrastive learning of tabular data: A data-centric benchmarking. *International Journal of Data Science and Analytics*, 20(4), 3069–3091. <https://doi.org/10.1007/s41060-024-00681-z>
- [24] Qu, J., Holzmüller, D., Varoquaux, G., & Le Morvan, M. T. (2025). A tabular foundation model for in-context learning on large data. In *International Conference on Machine Learning*, 50817–50847.
- [25] Chang, Y. H., Chen, F. C., & Lee, C. I. (2025). Developing an early warning system with personalized interventions to enhance academic outcomes for at-risk students in Taiwanese higher education. *Education Sciences*, 15(10), 1321. <https://doi.org/10.3390/educsci15101321>
- [26] Roslan, M. B., & Chen, C. (2022). Educational data mining for student performance prediction: A systematic literature review (2015–2021). *International Journal of Emerging Technologies in Learning*, 17(5), 147–179.
- [27] Somepalli, G., Goldblum, M., Schwarzschild, A., Bruss, C. B., & Goldstein, T. (2021). Saint: Improved neural networks for tabular data via row attention and contrastive pre-training. *arXiv*. <https://doi.org/10.48550/arXiv.2106.01342>
- [28] Angeioplastis, A., Aliprantis, J., Konstantakis, M., & Tsimpiris, A. (2025). Predicting student performance and enhancing learning outcomes: A data-driven approach using educational data mining techniques. *Computers*, 14(3), 83. <https://doi.org/10.3390/computers14030083>
- [29] Gorishniy, Y., Kotelnikov, A., & Babenko, A. (2025). Tabm: Advancing tabular deep learning with parameter-efficient ensembling. In *International Conference on Learning Representations*, 77899–77935. 2025
- [30] Hollmann, N., Müller, S., Purucker, L., Krishnakumar, A., Körfer, M., Hoo, S. B., . . . & Hutter, F. (2025). Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045), 319–326. <https://doi.org/10.1038/s41586-024-08328-6>

- [31] Erickson, N., Purucker, L., Tschalzev, A., Holzmüller, D., Desai, P., Salinas, D., & Hutter, F. (2026). Tabarena: A living benchmark for machine learning on tabular data. *Advances in Neural Information Processing Systems*, 38. <https://doi.org/10.52202/085713-0519>

How to Cite: Yermek, T. B., Yergesh, B., Yelemessov, A., Yelemessova, M., Bekmanova, G., Omarbekova, A., ..., & Lamasheva, Z. (2026). A Multi-Model Approach to Academic Performance Prediction Using Student–Semester Educational Data. *Artificial Intelligence and Applications*. <https://doi.org/10.47852/bonviewAIA620210406>

Appendix A. Attributes of the source LMS extract

The source extract is the JournalData sheet of the institutional grade journal, which contains 732,965 discipline-level records for 20,868 students across 14 academic years. Table A1 documents each source attribute; the derived student–semester variables computed from them are defined in Table 6 of Section 3.5.

Table A1
Attributes of the source LMS extract and their role in the modeling pipeline

Attribute (source field)	Type	Admissible values / range	Missing values	Role in the pipeline
StudentID	Categorical (identifier)	20,868 distinct values	Removed before aggregation if missing	Grouping/alignment key only; excluded from predictors and reported outputs
Academic Year	Text (temporal)	14 distinct academic years	Removed before aggregation if missing	Categorical predictor; parsed to start_year for temporal partitioning
Term	Integer (ordinal)	{1, 2} retained; –1 = non-standard (4,390 rows)	Removed before aggregation if missing	Semester ordering/aggregation; –1 excluded; not a predictor
Subject	Text (categorical)	8,153 distinct disciplines	0 (0%)	Counted per student–semester to obtain n_subjects
Mark	Numeric (continuous)	0–100; mean 87.16; SD 11.48	0 (0%)	Aggregated to avg/std/min/max; count gives n_marks; source of next-semester target

StudentID was retained only as a technical grouping key during longitudinal aggregation and target construction and was excluded from the model feature matrix and reported outputs. No personal, demographic, behavioural or attendance attributes are used as predictive features, which is the basis of the feature-availability limitation discussed in Section 5. In the verified JournalData extract, Subject and Mark contain no missing values. The preparation code nevertheless applies listwise deletion if StudentID, Academic Year, Term or Mark is missing before aggregation. Where a student has a single mark in a semester, std_mark is undefined and is set to zero.